Deterministic Approximate Counting of Polynomial Threshold Functions via a Derandomized Regularity Lemma

Rocco A. Servedio
Columbia University, New York, NY, USA

Li-Yang Tan
Stanford University, CA, USA

Abstract

We study the problem of deterministically approximating the number of satisfying assignments of a polynomial threshold function (PTF) over Boolean space. We present and analyze a scheme for transforming such algorithms for PTFs over *Gaussian space* into algorithms for the more challenging and more standard setting of Boolean space. Applying this transformation to existing algorithms for Gaussian space leads to new algorithms for Boolean space that improve on prior state-of-the-art results due to Meka and Zuckerman [19] and Kane [13]. Our approach is based on a bias-preserving derandomization of Meka and Zuckerman's regularity lemma for polynomials [19] using the [23] pseudorandom generator for PTFs.

2012 ACM Subject Classification Theory of computation → Pseudorandomness and derandomization

Keywords and phrases Derandomization, Polynomial threshold functions, deterministic approximate counting

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2021.37

Category RANDOM

Funding *Rocco A. Servedio*: NSF grants CCF-1814873, IIS-1838154, CCF-1563155, and Simons Collaboration on Algorithms and Geometry.
Li-Yang Tan: NSF CAREER award CCF-1942123.

Acknowledgements This material is based upon work supported by the National Science Foundation under grant numbers listed above. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation (NSF).

1 Introduction

Unconditional derandomization has emerged as a major topic of inquiry in complexity theory over the past several decades. One important strand in this study is the development of deterministic algorithms that can perform *approximate counting* for various function classes: given the description of a function $f \in \mathcal{C}$ and an accuracy parameter $\epsilon > 0$, deterministically output an estimate of the acceptance probability of f (i.e. $\mathbf{Pr}_{\boldsymbol{x} \leftarrow \{-1,1\}^n}[f(\boldsymbol{x}) = 1]$) that is additively accurate to within $\pm\epsilon$. This problem is trivially easy to solve with a randomized algorithm, but is much more challenging if a deterministic algorithm is required. Indeed, recall that the P vs. BPP problem is essentially equivalent to solving the deterministic approximate counting problem for $\mathcal{C}$ being the class of all polynomial-size circuits (and $\epsilon = 0.1$).

In this work we focus on the class of *low-degree polynomial threshold functions*, an important class of functions that has been the subject of intensive study in unconditional derandomization in recent years [5, 12, 11, 13, 16, 19, 14, 3, 4, 15, 9, 10, 17, 23, 22].

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2021).
Editors: Mary Wootters and Laura Sanità; Article No. 37; pp. 37:1–37:18
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1.1 Background: Deterministic approximate counting algorithms for PTFs

A *degree-d polynomial threshold function* (PTF) is a function $f(x) = \mathrm{sign}(p(x))$ where $p(x_1, \dots, x_n)$ is a polynomial of degree at most d and $\mathrm{sign} : \mathbb{R} \to \{-1, 1\}$ outputs 1 iff its argument is nonnegative. Deterministic approximate counting algorithms for PTFs have been well studied in a number of different works, and the following table summarizes the runtimes of the fastest known algorithms prior to this work:

Table 1 Prior work on deterministic approximate counting algorithms for degree-d PTFs. In the runtime of [13], $O_d(\cdot)$ hides an Ackermann-type dependence on d, and likewise, in the runtime of [4], $O_{d,\epsilon}(\cdot)$ hides an Ackermann-type dependence on d and $1/\epsilon$. With the exception of [4], all these algorithms are based on the construction of pseudorandom generators for PTFs.

Reference	Runtime
[19]	$n^{(d/\epsilon)^{O(d)}}$
[13]	$n^{O_d(\mathrm{poly}(1/\epsilon))}$
[4]	$O_{d,\epsilon}(1) \cdot n^{O(d)}$
[17]	$\mathrm{poly}(n) \cdot \exp(2^{\tilde{O}(\sqrt{\log(1/\epsilon)})})$ for $d = 2$
[23]	$\exp(2^{\sqrt{d \log n}}) \cdot \mathrm{quasipoly}(1/\epsilon)$

1.1.1 Algorithms for PTFs over Gaussian space

A fruitful theme that has emerged in the study of PTFs concerns the relationship between PTFs in the standard setting of *Boolean space* – PTFs over $\{-1, 1\}^n$ endowed with the uniform distribution – and PTFs in the setting of *Gaussian space*: PTFs over $\mathbb{R}^n$ endowed with the Gaussian measure $N(0, 1)^n$.

The Gaussian setting enjoys numerous useful features that are not afforded by the Boolean setting, most of them owing to the continuous nature of $\mathbb{R}^n$ and the rotational invariance of the Gaussian measure. In fact, the problem of approximate counting of degree-d PTFs over Gaussian space (i.e. approximating $\mathbf{Pr}_{\boldsymbol{g} \leftarrow N(0,1)^n}[f(\boldsymbol{g}) = 1]$ where f is a degree-d PTF) can be seen to be a *special case* of the same problem over Boolean space, in the sense that an algorithm for the latter setting can be used to obtain an algorithm with a comparable runtime for the former setting. (This is a consequence of the invariance principle [20].) For this reason, many works have focused on the special case of designing deterministic approximate counting algorithms for PTFs over Gaussian space. There are by now a number of results in this setting for which no counterparts are yet known for the more challenging Boolean setting:

Contrasting the state of the art over Boolean and Gaussian space. Comparing Tables 1 and 2, we note that the runtime of [12] is strictly better than those of [19]'s and [13]'s algorithms for the Boolean setting; the runtime of [14] is strictly better than that of [13]; the runtime of [3] is strictly better than that of [17]; and the runtime of [22] remains subexponential for $d = 2^{\tilde{\Omega}(\sqrt{\log n})}$, whereas all algorithms for the Boolean setting trivialize once $d = \Omega(\log n)$.

Table 2 Current best deterministic approximate counting algorithms for degree-d PTFs over *Gaussian space*. With the exception of [3], all these algorithms are based on the construction of pseudorandom generators for PTFs over Gaussian space.

Reference	Running time
[12]	$n^{2^{O(d)} \cdot \mathrm{poly}(1/\epsilon)}$
[14]	$n^{O_{d,\kappa}(1/\epsilon)^{\kappa}}$ for all $\kappa > 0$
[3]	$\mathrm{poly}(n, 1/\epsilon)$ for $d = 2$
[22]	$n^{(d/\epsilon)^{O(\log d)}}$

1.2 This work: Upgrading algorithms for Gaussian space into ones for Boolean space

In this work we establish a new connection between the derandomization of PTFs over Boolean and Gaussian space. We leverage this connection to transform existing deterministic approximate counting algorithms for the Gaussian setting (i.e. those summarized in Table 2) into new state-of-the-art deterministic algorithms for approximate counting of PTFs for the more challenging Boolean setting, improving upon those summarized in Table 1.

The runtimes of our new algorithms improve upon the prior state of the art for a broad range of parameters. For $d = \Theta(1)$, we obtain a strict improvement for all ϵ satisfying $2^{-\Theta(\sqrt{\log n})} \leq \epsilon \leq o_n(1)$. For $d = \omega_n(1)$, we obtain a strict improvement for all ϵ satisfying $d \log(d/\epsilon) \leq \Theta(\log n)$. We now give precise statements of the runtimes of our new algorithms, and provide example parameter settings that highlight the main qualitative advantages of these new runtimes.

1.3 Our results: New deterministic approximate counting algorithms for PTFs over Boolean space

First, by instantiating our framework with [12]'s algorithm for the Gaussian setting, we obtain the following algorithm for the Boolean setting:

▶ **Theorem 1.** *There is a deterministic algorithm for ϵ-approximate counting n-variable degree-d PTFs over Boolean space that runs in time*

$$\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right) \cdot n^{2^{O(d)} \cdot \mathrm{poly}(1/\epsilon)}.$$

This runtime is a strict improvement of [19]'s runtime and very nearly matches the $n^{2^{O(d)} \cdot \mathrm{poly}(1/\epsilon)}$ running time of the [12] algorithm for Gaussian space. For any $\epsilon = \Theta(1/\mathrm{polylog}(n))$, our runtime remains $n^{\mathrm{polylog}(n)}$ for d as large as $\tilde{\Omega}(\log\log n)$, whereas the runtimes of all previous algorithms for the Boolean setting exceed $n^{\mathrm{polylog}(n)}$ once d is even a slightly superconstant function of n.

Next, by instantiating our framework with [14]'s algorithm for the Gaussian setting, we obtain the following algorithm for the Boolean setting:

▶ **Theorem 2.** *For all $\kappa > 0$, there is a deterministic algorithm for ϵ-approximate counting n-variable degree-d PTFs over Boolean space that runs in time*

$$n^{O_{d,\kappa}(1/\epsilon)^{\kappa}}.$$

This runtime is a strict improvement of [13]'s runtime and matches that of [14]'s algorithm for the Gaussian setting. For arbitrarily large constants $c, d \in \mathbb{N}$ and $\epsilon = 1/(\log n)^c$, our runtime is barely superpolynomial, $n^{O((\log n)^{\kappa})}$ for any arbitrarily small constant $\kappa > 0$, whereas all previous algorithms for the Boolean setting run in time at least $n^{(\log n)^{\Omega(c)}}$ or $n^{(\log n)^{\Omega(d)}}$.

Table 3 Our new algorithms for deterministic approximate counting of degree-d PTFs over Boolean space. The runtime of Theorem 1 is a strict improvement of [19]'s; the runtime of Theorem 2 is a strict improvement of [13]'s, and matches that of [14]'s algorithm for Gaussian space.

	Runtime	Follows by instantiating our framework with:
Theorem 1	$\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right) \cdot n^{2^{O(d)} \cdot \mathrm{poly}(1/\epsilon)}$	[12]'s Gaussian PRG
Theorem 2	$n^{O_{d,\kappa}(1/\epsilon)^{\kappa}}$	[14]'s Gaussian PRG

Finally, we remark that:

- Our framework can also be instantiated with [3]'s algorithm for degree-2 PTFs in the Gaussian setting to recover [17]'s PRG-based algorithm for degree-2 PTFs in the Boolean setting (and in fact, we are able to improve it slightly by eliminating the polylog$(1/\epsilon)$ factor suppressed by the $\tilde{O}(\cdot)$ in its runtime);
- Our framework can also be instantiated with [22]'s algorithm for degree-d PTFs in the Gaussian setting, yielding a deterministic algorithm for degree-d PTFs over Boolean space that runs in time $\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right) \cdot n^{(d/\epsilon)^{O(\log d)}}$. Like Theorem 1, this runtime is a strict improvement of [19]'s runtime.

1.4 Our approach: Derandomizing Meka and Zuckerman's regularity lemma

Our main new tool is a *derandomization of the [19] regularity lemma*. To explain what this means, we begin by recalling the basics of the original [19] regularity lemma.

A multivariate polynomial p is said to be *regular* if, intuitively, no variable has high influence (we give precise definitions in Section 2). Let us recall the original [19] regularity lemma: given any degree-d polynomial over $\{-1,1\}^n$, it gives an efficient (deterministic) algorithm which builds a not-too-large decision tree such that at almost every leaf ρ, the resulting polynomial p_ρ (p restricted according the partial assignment corresponding to that leaf) is such that either (i) p_ρ is regular, or (ii) sign(p_ρ) is close to either the constant $+1$ function or the constant -1 function.[1]

[1] We note that a similar regularity lemma was given in simultaneous work of [6] (see also [18]); that work employed a slightly different technical definition of what it means for a polynomial to be "regular", and it gave a similar algorithm to build a decision tree with similar properties for that related notion. However, in the current work for technical reasons it is essential that we use the [19] notion of regularity; we explain this in more detail in Remark 22 in Appendix B.2.

The [19] regularity lemma is useful for derandomization because it lets one reduce the general Boolean case to the "regular" Boolean case, which can be easier because sophisticated mathematical tools like central limit theorems and invariance principles can be brought to bear on regular polynomials over $\{-1,1\}^n$ to relate them to the corresponding polynomials over the Gaussian domain. Indeed, the [19] regularity lemma and related results play an important role in a number of PRG and approximate counting results for polynomial threshold functions, including the works of [19, 13, 4] mentioned above. However there is often a substantial algorithmic cost associated with the use of a regularity lemma, because building the decision tree (or equivalently, exploring all of its leaves) can be relatively expensive.

Our derandomization of the regularity lemma. The above-described standard strategy of building and visiting all of the leaves of a decision tree corresponds to using true uniform randomness to choose a path through the decision tree. The intuition behind our derandomized version of the regularity lemma is as follows: by choosing a path through the decision tree according to a suitable *pseudorandom* distribution, it is possible, from an algorithmic perspective, to "build and visit only a tiny fraction of the leaves of the decision tree." This can be much more efficient than visiting all leaves.

Intuitively, the leaves that our derandomized regularity lemma constructs are determined by the output of a PRG for degree-d PTFs over m variables where m is the depth of the decision tree.[2] As our analysis shows, for the purpose of deterministic approximate counting for the original PTF $\mathrm{sign}(p)$, it suffices to do deterministic approximate counting on just the PTFs $\mathrm{sign}(p_\rho)$ for these (relatively few) leaves ρ.

More precisely, we prove a general result, Theorem 15, which encapsulates the above approach. It outputs a collection of restrictions (which can be thought of as a very small subset of the leaves of the decision tree that the original regularity lemma constructs) with the following property: given an accurate estimate of the fraction of assignments satisfying $\mathrm{sign}(p_\rho)$ for each restriction ρ in the collection, combining these estimates in the obvious way gives an accurate estimate of the overall fraction of inputs in $\{-1,1\}^n$ that satisfy the original PTF $\mathrm{sign}(p)$. Moreover (and crucially), each restriction ρ in the collection is such that either the restricted polynomial p_ρ is highly regular, or else $\mathrm{sign}(p_\rho)$ is a close-to-constant function.

For the purpose of deterministic approximate counting, restrictions where $\mathrm{sign}(p_\rho)$ is a close-to-constant function are easy to handle, and thanks to the invariance principle, at restrictions where p_ρ is regular we can do Gaussian deterministic approximate counting and the resulting estimate of $\mathbf{Pr}_{\boldsymbol{g}\leftarrow N(0,1)^n}[\mathrm{sign}(p_\rho(\boldsymbol{g}))=1]$ will be an accurate estimate of $\mathbf{Pr}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p_\rho(\boldsymbol{x}))]$ for the Boolean problem. Thus, the overall running time of the deterministic approximate counting algorithm we obtain from the regularity lemma is essentially the running time of (i) "fooling Boolean PTFs over few variables" (to build the tree) times the running time of (ii) "Gaussian determinstic approximate counting" (to handle the regular leaves).

1.4.1 Applying the derandomized regularity lemma

To obtain an efficient deterministic approximate counting algorithm from this approach, in part (i) above it is okay to use a PRG with a relatively poor dependence on the number of variables, since the number of variables is quite small. Such a generator is provided for us by

[2] This is actually an oversimplification: the regularity lemma works in a sequence of "atomic stages" to build a tree, and our approach actually works by derandomizing each atomic stage separately. The cost of a single atomic stage provides the dominant contribution to the overall cost, though, so the intuition is correct.

the [23] PRG (or rather a slight variant of it which we need for technical reasons); we can afford this generator's poor dependence on the number of variables, and using it lets us take advantage of its better dependence on the other parameters (it is clear from Table 1 that [23] has a better dependence on both d and ϵ than any of the other algorithms in that table).

By applying our derandomized regularity lemma with (essentially) the [23] PRG for part (i) and the Gaussian result of [12] for part (ii), we obtain Theorem 1. By using instead [14]'s algorithm for the Gaussian setting for part (ii), we obtain Theorem 2.

2 Preliminaries

We start by establishing some basic notation. We write $[n]$ to denote $\{1, 2, \ldots, n\}$ and $[k, \ell]$ to denote $\{k, k+1, \ldots, \ell\}$. We use bold font to denote random variables. We write $\mathbf{E}[\boldsymbol{X}]$ and $\mathbf{Var}[\boldsymbol{X}]$ to denote expectation and variance of a random variable $\boldsymbol{X}$ and write $\mathbf{E}_{\boldsymbol{X} \leftarrow \mathcal{D}}[\boldsymbol{X}]$, $\mathbf{Var}_{\boldsymbol{X} \leftarrow \mathcal{D}}[\boldsymbol{X}]$, and the like to indicate that the random variable $\boldsymbol{X}$ has distribution $\mathcal{D}$. If S is a finite set then "$\boldsymbol{X} \leftarrow S$" means that $\boldsymbol{X}$ is distributed uniformly over S; if no distribution is specified for a random variable taking values in $\{-1,1\}^n$ then the implied distribution is uniform over $\{-1,1\}^n$. For $x \in \{-1,1\}^n$ and $A \subseteq [n]$ we write x_A to denote $(x_i)_{i \in A}$.

For a function $f : \{-1,1\}^n \to \mathbb{R}$ and $q \geq 1$, we denote by $\|f\|_q$ its ℓ_q norm with respect to the uniform distribution, i.e., $\|f\|_q \stackrel{\text{def}}{=} \mathbf{E}[|p(\boldsymbol{x})|^q]^{1/q} = \mathbf{E}_{\boldsymbol{x} \leftarrow \{-1,1\}^n}[|p(\boldsymbol{x})|^q]^{1/q}$. We write $\|f\|_{q,\mathcal{D}}$ to denote its ℓ_q norm with respect to the distribution $\mathcal{D}$, i.e. $\|f\|_{q,\mathcal{D}} \stackrel{\text{def}}{=} \mathbf{E}_{\boldsymbol{x} \leftarrow \mathcal{D}}[|p(\boldsymbol{x})|^q]^{1/q}$.

For Boolean-valued functions $f, g : \{-1,1\}^n \to \{-1,1\}$ the distance between f and g, denoted $\operatorname{dist}(f,g)$, is $\mathbf{Pr}[f(\boldsymbol{x}) \neq g(\boldsymbol{x})]$.

2.1 Fourier analysis of Boolean functions

Fourier Analysis over $\{-1,1\}^n$ and Influences. We consider functions $f : \{-1,1\}^n \to \mathbb{R}$, and we think of the inputs x to f as being distributed according to the uniform probability distribution. The set of such functions forms a 2^n-dimensional real inner product space with inner product given by $\langle f, g \rangle = \mathbf{E}[f(\boldsymbol{x})g(\boldsymbol{x})]$. The set of functions $(\chi_S)_{S \subseteq [n]}$ defined by $\chi_S(x) = \prod_{i \in S} x_i$ forms a complete orthonormal basis for this space. Given a function $f : \{-1,1\}^n \to \mathbb{R}$ we define its *Fourier coefficients* by $\widehat{f}(S) \stackrel{\text{def}}{=} \mathbf{E}[f(\boldsymbol{x})\chi_S(\boldsymbol{x})]$, and we have that $f(x) = \sum_S \widehat{f}(S)\chi_S(x)$. We refer to the maximum $|S|$ over all nonzero $\widehat{f}(S)$ as the *Fourier degree* of f.

As a consequence of orthonormality we have *Plancherel's identity* $\langle f, g \rangle = \sum_S \widehat{f}(S)\widehat{g}(S)$, which has as a special case *Parseval's identity*, $\mathbf{E}[f(\boldsymbol{x})^2] = \sum_S \widehat{f}(S)^2$. From this it follows that for every $f : \{-1,1\}^n \to \{-1,1\}$ we have $\sum_S \widehat{f}(S)^2 = 1$. We recall the well-known fact that the total influence $\operatorname{Inf}(f)$ of any Boolean function equals $\sum_S \widehat{f}(S)^2|S|$. Note that, in this setting, the expectation and the variance can be expressed in terms of the Fourier coefficients of f by $\mathbf{E}[f] = \widehat{f}(\emptyset)$ and $\mathbf{Var}[f] = \sum_{\emptyset \neq S \subseteq [n]} \widehat{f}(S)^2$.

Let $f : \{-1,1\}^n \to \mathbb{R}$ and $f(x) = \sum_S \widehat{f}(S)\chi_S(x)$ be its Fourier expansion. The *influence* of variable i on f is $\operatorname{Inf}_i(f) \stackrel{\text{def}}{=} \sum_{S \ni i} \widehat{f}(S)^2$, and the *total influence* of f is $\operatorname{Inf}(f) = \sum_{i=1}^n \operatorname{Inf}_i(f)$.

Bounded independence and bounded uniformity distributions. A distribution $\mathcal{D}$ on $\{-1,1\}^n$ is said to be *k-wise independent* if any collection of k distinct coordinates $i_1, \ldots, i_k$ are such that $\boldsymbol{x}_{i_1}, \ldots, \boldsymbol{x}_{i_k}$ are independent when $\boldsymbol{x} \leftarrow \mathcal{D}$. If each $\boldsymbol{x}_i$ additionally is uniform over $\{-1,1\}$, then the distribution $\mathcal{D}$ is said to be *k-wise uniform.* We note that if $\mathcal{D}$ is a k-wise uniform distribution over $\{-1,1\}^n$, then for any degree-k polynomial p, it holds by linearity of expectation that

$$\mathop{\mathbf{E}}_{\boldsymbol{z}\leftarrow\mathcal{D}}[p(\boldsymbol{z})] = \mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[p(\boldsymbol{x})], \tag{1}$$

and hence if $\mathcal{D}$ is (qk)-wise uniform for even q, it holds that

$$\|p\|_q = \|p\|_{q,\mathcal{D}}. \tag{2}$$

Useful probability bounds. We first recall the (2,4)-Hypercontractivity theorem of [2, 7]:

▶ **Theorem 3** ((2,4)-Hypercontractivity, special case of Theorem 9.21 of [21] / Lemma 4.3 of [8])**.** *Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial. Then*

$$\|p\|_4 \leq 3^{d/2} \cdot \|p\|_2.$$

For our purposes we will need a *derandomized* version of Theorem 3, where the expectations are with respect to a suitable pseudorandom distribution. As an immediate consequence of Equation (2), we obtain the following corollary of Theorem 3:

▶ **Corollary 4** ((2,4)-Hypercontractivity for bounded-uniformity distributions)**.** *Let p : $\{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial. If $\mathcal{D}$ is a $4d$-wise uniform distribution over $\{-1,1\}^n$, then*

$$\|p\|_{4,\mathcal{D}} \leq 3^{d/2} \cdot \|p\|_{2,\mathcal{D}}.$$

We will need the following fact, which is a consequence of $(2,4)$-hypercontractivity and states that a low-degree polynomial must exceed its expectation with nonnegligible probability:

▶ **Fact 5** (Lemma 5.4 of [8])**.** *Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial normalized so that $\mathbf{Var}[p] = 1$. Then there is an absolute constant $C > 0$ such that*

$$\mathop{\mathbf{Pr}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}\left[p(\boldsymbol{x}) \geq \mathbf{E}[p] + 2^{-Cd}\right] \geq 2^{-O(d)}.$$

We will also need a derandomized version of Fact 5:

▶ **Fact 6** (Fact 5 for bounded-uniformity distributions)**.** *Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial normalized so that $\mathbf{Var}[p] = 1$. If $\mathcal{D}$ is a $4d$-wise uniform distribution over $\{-1,1\}^n$, then there is an absolute constant $C > 0$ such that*

$$\mathop{\mathbf{Pr}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}\left[p(\boldsymbol{x}) \geq \mathbf{E}[p] + 2^{-Cd}\right] \geq 2^{-O(d)}.$$

For the sake of completeness, we include the short proof of Fact 6 in Appendix A.

Invariance. We recall the invariance principle of Mossel, O'Donnell and Oleszkiewicz, specifically Theorem 3.19 under hypothesis **H4** in [20]:

▶ **Theorem 7** ([20])**.** *Let $p(x) = \sum_{S\subseteq[n],|S|\leq d} \widehat{p}(S)\chi_S(x)$ be a degree-d multilinear polynomial with $\mathbf{Var}[p] = 1$. Suppose each coordinate $i \in [n]$ has $\mathrm{Inf}_i(p) \leq \tau$. Then,*

$$\sup_{t\in\mathbb{R}}\left|\mathop{\mathbf{Pr}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[p(\boldsymbol{x}) \leq t] - \mathop{\mathbf{Pr}}_{\mathcal{G}\leftarrow N(0,1)^n}[p(\mathcal{G}) \leq t]\right| \leq O(d\tau^{1/(8d)}).$$

2.2 PTFs, regularity, and the critical index

▶ **Definition 8** (Regularity). *We say that a degree-d polynomial p is τ-regular if*

$$\sqrt{\sum_{j=i}^{n} \mathrm{Inf}_i(p)^2} \leq \tau \sum_{j=1}^{n} \mathrm{Inf}_j(p).$$

▶ **Definition 9** (τ-critical index). *Let p be a degree-d polynomial, and assume (without loss of generality) that the variables of p are ordered so that $\mathrm{Inf}_1(p) \geq \mathrm{Inf}_2(p) \geq \ldots \geq \mathrm{Inf}_n(p)$. The τ-critical index of p is the least i such that*

$$\mathrm{Inf}_{i+1}(p) \leq \tau^2 \sum_{j=i+1}^{n} \mathrm{Inf}_j(p).$$

3 Derandomizing Meka and Zuckerman's regularity lemma

3.1 Overview of [19]'s regularity lemma and its proof

In this subsection we state Meka and Zuckerman's regularity lemma for low-degree polynomials [19], and recall its key elements at a level of detail which will enable us to build on those elements.

▶ **Lemma 10** (Implicit in the proof of Lemma 5.17 of [19]). *There is a deterministic algorithm which, on input a degree-d polynomial p and parameters τ, ϵ, δ, outputs a decision tree of depth*

$$\mathrm{depth}(d, \tau, \epsilon, \delta) := \frac{2^{O(d)}}{\tau^2} \cdot \log(\tfrac{1}{\delta}) \log(\tfrac{1}{\epsilon})$$

with the following property: with probability $1 - \epsilon$, a random path down the tree reaches a leaf $\boldsymbol{\rho}$ such that $p_{\boldsymbol{\rho}}$ is either

1. *τ-regular, or*
2. *the PTF $\mathrm{sign}(p_{\boldsymbol{\rho}})$ is δ-close to the constant function $\mathrm{sign}(\mathbf{E}[p_{\boldsymbol{\rho}}])$.*

The running time of this tree construction algorithm is $\mathrm{poly}(n^d, 2^{\mathrm{depth}(d,\tau,\epsilon,\delta)})$.

The algorithm of [19] recursively constructs the tree in a sequence of simple "atomic steps". We now describe how a single atomic step works. Consider a leaf ρ of the decision tree; initially the leaf ρ is simply the root of the tree corresponding to the empty restriction. The algorithm behaves differently depending on how large the τ-critical index of p_ρ is:

Large critical index. If the polynomial p_ρ has "large" τ-critical index (larger than a parameter K which is $2^{O(d)} \log(1/\delta)/\tau^2$) then an "atomic step" consists of fixing the K variables which have the highest influence in p_ρ, i.e. replacing the current leaf with a complete depth-K decision tree that exhaustively queries those variables. The key to the analysis of this case is the following structural result, which is Lemma 5.2 of [8]:

▶ **Lemma 11** (restatement of [8]'s Lemma 5.2: Large critical index). *There is a universal constant $C_1 > 0$ such that the following holds. Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial with τ-critical index at least $K := 2^{C_1 d} \log(1/\delta)/\tau^2$. Then*

$$\Pr_{\boldsymbol{\rho} \leftarrow \{-1,1\}^{[K]}} \left[\mathbf{Var}[p_{\boldsymbol{\rho}}] \leq \delta\, \mathbf{E}[p_{\boldsymbol{\rho}}]^2 \right] \geq 2^{-O(d)},$$

and consequently by Chebyshev's inequality,

$$\Pr_{\boldsymbol{\rho} \leftarrow \{-1,1\}^{[K]}} \left[\mathrm{sign}(p_{\boldsymbol{\rho}}) \textit{ is } \delta\textit{-close to } \mathrm{sign}(\mathbf{E}[p_{\boldsymbol{\rho}}]) \right] \geq 2^{-O(d)}.$$

Small critical index. If the polynomial has small τ-critical index (smaller than K), then an "atomic step" consists of fixing the "head" variables $[k]$ up to the critical index (again building a complete decision tree over those $k \le K$ variables). The key to the analysis of this case is the following structural result, which is Lemma 5.1 of [8]:

▶ **Lemma 12** (restatement of [8]'s Lemma 5.1: Small critical index). *Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial with τ-critical index $k \in [n]$. Then*

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho} \leftarrow \{-1,1\}^{[k]}} \left[p_{\boldsymbol{\rho}} \textit{ is } \tau'\textit{-regular} \right] \ge 2^{-O(d)}, \quad \textit{where } \tau' \le 2^{O(d)} \cdot \tau.$$

Given these two results, a relatively straightforward analysis (which is in fact a special case of the analysis we give in the subsequent subsections) shows that after at most $2^{O(d)} \log(1/\epsilon)$ levels of these "atomic steps", at most an ϵ fraction of paths will not have terminated either in a close-to-constant leaf or a regular leaf.

3.2 The high level idea of our approach: Derandomizing each "atomic step" in a bias-preserving manner

The first important technical ingredient of our approach is in the following two lemmas, Lemma 13 and 14, which give *derandomized* versions of Lemma 11 and 12 respectively. Intuitively, these results say that in each of Lemma 11 and 12, rather than considering the uniform distribution over all restrictions fixing $[K]$ and $[k]$ respectively, it suffices to consider instead a suitable *pseudorandom* distribution over restrictions.

▶ **Lemma 13** (Bounded uniformity suffices for Lemma 11: Large critical index). *Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial with τ-critical index at least $K := 2^{C_1 d} \log(1/\delta)/\tau^2$, where C_1 is the universal constant from Lemma 11. Let $\mathcal{D}$ be a $4d$-wise uniform distribution over $\{-1,1\}^K$. Then*

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho} \leftarrow \mathcal{D}} \left[\mathbf{Var}[p_{\boldsymbol{\rho}}] \le \delta\, \mathbf{E}[p_{\boldsymbol{\rho}}]^2 \right] \ge 2^{-O(d)},$$

and consequently by Chebyshev's inequality,

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho} \leftarrow \mathcal{D}} \left[\mathrm{sign}(p_{\boldsymbol{\rho}}) \textit{ is } \delta\textit{-close to } \mathrm{sign}(\mathbf{E}[p_{\boldsymbol{\rho}}]) \right] \ge 2^{-O(d)}.$$

▶ **Lemma 14** (Bounded uniformity suffices for Lemma 12: Small critical index). *Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial with τ-critical index $k \in [n]$. Let $\mathcal{D}$ be a $4d$-wise uniform distribution over $\{-1,1\}^k$. Then*

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho} \leftarrow \mathcal{D}} \left[p_{\boldsymbol{\rho}} \textit{ is } \tau'\textit{-regular} \right] \ge 2^{-O(d)}, \quad \textit{where } \tau' \le 2^{O(d)} \cdot \tau.$$

We prove Lemma 13 in Appendix B.1 and prove Lemma 14 in Appendix B.2. Combining these results with a PRG that fools degree-d PTFs over m variables (where "m" should be thought of as $\ll n$), we establish our bias-preserving derandomized regularity lemma:

▶ **Theorem 15** (Bias-preserving derandomization of [19]'s regularity lemma, Lemma 10). *Let $\mathcal{G}_{\mathrm{PTF}}$ be a PRG with seed length $s(m, d, \eta)$ that*

(i) *is $4d$-wise uniform and*

(ii) *η-fools degree-d PTFs over m variables.*

There is a deterministic algorithm Build-Restrictions *which, on input a degree-d polynomial $p : \{-1,1\}^n \to \mathbb{R}$ and parameters ϵ, δ, and τ, outputs a collection $\mathcal{R}$ of restrictions,*

$$|\mathcal{R}| \leq \exp(s(m,d,\eta) \cdot 2^{O(d)} \log(\tfrac{1}{\epsilon}))$$

where

$$m \leq \frac{2^{O(d)}}{\tau^2} \log(\tfrac{1}{\delta}) \quad \text{and} \quad \eta = \frac{\epsilon}{2^{O(d)} \log(\frac{1}{\epsilon})}$$

with the following property: with probability $1-\epsilon$ over a draw $\boldsymbol{\rho} \leftarrow \mathcal{R}$, the restricted polynomial $p_{\boldsymbol{\rho}}$ is either

1. *τ-regular, or*
2. *the PTF $\mathrm{sign}(p_{\boldsymbol{\rho}})$ is δ-close to the constant function $\mathrm{sign}(\mathbf{E}[p_{\boldsymbol{\rho}}])$.*

Furthermore the collection of restrictions $\mathcal{R}$ is bias-preserving, in the sense that

$$\left| \mathop{\mathbf{E}}_{\boldsymbol{\rho} \leftarrow \mathcal{R}} \Big[\mathop{\mathbf{E}}_{\boldsymbol{x} \leftarrow \{-1,1\}^n} [\mathrm{sign}(p_{\boldsymbol{\rho}}(\boldsymbol{x}))] \Big] - \mathop{\mathbf{E}}_{\boldsymbol{x} \leftarrow \{-1,1\}^n} [\mathrm{sign}(p(\boldsymbol{x}))] \right| \leq \epsilon. \tag{3}$$

The running time of Build-Restrictions *is* $\mathrm{poly}(n^d, |\mathcal{R}|)$.

We prove Theorem 15 in Section 3.3. At a high level, the argument is an extension of the analysis that establishes the original regularity lemma of [19] from Lemma 11 and 12.

In Section 4 we apply Theorem 15 to obtain new deterministic approximate counting results for degree-d Boolean PTFs. We do this by instantiating the pseudorandom generator $\mathcal{G}_{\mathrm{PTF}}$ using (a slight variant of) the [23] pseudorandom generator, and by using invariance principles and pseudorandom generators for Gaussian PTFs to obtain the required estimates of $\mathbf{E}_{\boldsymbol{x} \leftarrow \{-1,1\}^n}[\mathrm{sign}(p_\rho)]$ for the regular polynomials p_ρ.

3.3 Proof of Theorem 15: Bias-preserving derandomization of [19]'s regularity lemma

We start with a simple fact about bias preservation:

▶ **Fact 16** (Bias preservation). *Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d polynomial and let $H \sqcup T$ be a partition of $[n]$ into two disjoint sets. Let $\mathcal{D}$ be a pseudorandom distribution over $\{-1,1\}^H$ that η-fools degree-d PTFs over the variables in H. Then*

$$\left| \mathop{\mathbf{E}}_{\substack{\boldsymbol{x} \leftarrow \{-1,1\}^H \\ \boldsymbol{y} \leftarrow \{-1,1\}^T}} [\mathrm{sign}(p(\boldsymbol{x},\boldsymbol{y}))] - \mathop{\mathbf{E}}_{\substack{\boldsymbol{z} \leftarrow \mathcal{D} \\ \boldsymbol{y} \leftarrow \{-1,1\}^T}} [\mathrm{sign}(p(\boldsymbol{z},\boldsymbol{y}))] \right| \leq \eta. \tag{4}$$

Proof. This follows directly from the fact that for any fixed outcome $y \in \{-1,1\}^T$, the function $\mathrm{sign}(p_y(x)) = \mathrm{sign}(p(x,y))$ is a degree-d PTF over the variables in H (i.e. the class of degree-d PTFs is closed under restrictions). ◀

The following will be the key subroutine for our algorithm:

▶ **Lemma 17** (Single atomic step). *Let $\mathcal{G}_{\mathrm{PTF}}$ be a PRG with seed length $s(m,d,\eta)$ that*

(i) *is $4d$-wise uniform and*

(ii) *η-fools degree-d PTFs over m variables.*

There is a universal constant $C_2 > 0$ such that the following holds. There is a deterministic algorithm Build-Restrictions-Atomic *which, on input a degree-d polynomial $p : \{-1,1\}^n \to \mathbb{R}$ and parameters δ, η, and τ, outputs a collection $\mathcal{R}_{\mathrm{atomic}}(p)$ of restrictions,*

$$|\mathcal{R}_{\mathrm{atomic}}(p)| \le \exp(s(m,d,\eta)), \quad m \le \frac{2^{O(d)}}{\tau^2}\log(\tfrac{1}{\delta})$$

with the following property: with probability $\ge 2^{-C_2 d}$ over a draw $\boldsymbol{\rho} \leftarrow \mathcal{R}_{\mathrm{atomic}}(p)$, the restricted polynomial $p_{\boldsymbol{\rho}}$ is either

1. *τ-regular, or*
2. *satisfies $\mathbf{Var}[p_{\boldsymbol{\rho}}] \le \delta\,\mathbf{E}[p_{\boldsymbol{\rho}}]^2$, and consequently by Chebyshev's inequality, the PTF $\mathrm{sign}(p_{\boldsymbol{\rho}})$ is δ-close to the constant function $\mathrm{sign}(\mathbf{E}[p_{\boldsymbol{\rho}}])$.*

Furthermore, this collection of restrictions $\mathcal{R}_{\mathrm{atomic}}(p)$ is bias-preserving, in the sense that:

$$\left| \mathop{\mathbf{E}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}_{\mathrm{atomic}}(p)}\left[\mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p_{\boldsymbol{\rho}}(\boldsymbol{x}))]\right] - \mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p(\boldsymbol{x}))]\right| \le \eta. \tag{5}$$

The running time of Build-Restrictions-Atomic *is $\mathrm{poly}(n^d, |\mathcal{R}_{\mathrm{atomic}}(p)|)$.*

Proof. Define $\overline{\tau} := \tau \cdot 2^{-C_3 d}$ where $C_3 > 0$ is a universal constant that we will set later. The algorithm Build-Restrictions-Atomic begins by computing $\mathrm{Inf}_i(p)$ for all $i \in [n]$, which can be done deterministically in time $\mathrm{poly}(n^d)$ via the Fourier formula $\mathrm{Inf}_i(p) = \sum_{S\ni i}\hat{p}(S)^2$. With these values, the algorithm then determines whether the $\overline{\tau}$-critical index of p is large (i.e. at least $K := 2^{C_1 d}\log(1/\delta)/\overline{\tau}^2$ where C_1 is the universal constant from Lemma 11) or small (i.e. at most $k < K$). Let $H \subseteq [n]$ be the K most influential variables of p in the case where p has large $\overline{\tau}$-critical index and the k most influential ones otherwise, and let $T := [n] \setminus H$.

We define $\mathcal{R}_{\mathrm{atomic}}(p)$ to be the set of all restrictions $\rho \in \{-1,1\}^H \times \{*\}^T$ such that $\rho_H \in \mathrm{range}(\mathcal{G}_{\mathrm{PTF}})$, where $\mathcal{G}_{\mathrm{PTF}}$ is a PRG with seed length $s(|H|, d, \eta)$ that is $4d$-wise uniform and η-fools degree-d PTFs over $\{-1,1\}^H$. Note that the size of $\mathcal{R}_{\mathrm{atomic}}(p)$ is indeed as claimed in the statement of the lemma:

$$\begin{aligned}&|\mathcal{R}_{\mathrm{atomic}}(p)| \le \exp(s(|H|,d,\eta)),\\ &\text{where } |H| \le |\mathrm{range}(\mathcal{G}_{\mathrm{PTF}})| \le \frac{2^{O(d)}}{\overline{\tau}^2}\log(\tfrac{1}{\delta}) = \frac{2^{O(d)}}{\tau^2}\log(\tfrac{1}{\delta}).\end{aligned}$$

By the $4d$-uniformity of $\mathcal{G}_{\mathrm{PTF}}$ and our definition of H, it follows from Lemma 13 and 14 that with probability $\ge 2^{-O(d)}$ over a draw $\boldsymbol{\rho} \leftarrow \mathcal{R}_{\mathrm{atomic}}$, the restricted polynomial $p_{\boldsymbol{\rho}}$ is either

1. $(2^{O(d)} \cdot \overline{\tau})$-regular, or
2. satisfies $\mathbf{Var}[p_{\boldsymbol{\rho}}] \le \delta\,\mathbf{E}[p_{\boldsymbol{\rho}}]^2$, and consequently by Chebyshev's inequality, the PTF $\mathrm{sign}(p_{\boldsymbol{\rho}})$ is δ-close to the constant function $\mathrm{sign}(\mathbf{E}[p_{\boldsymbol{\rho}}])$.

We choose the universal constant C_3 to ensure that $2^{O(d)} \cdot \overline{\tau} = 2^{O(d)} \cdot \tau \cdot 2^{-C_3 d} \le \tau$. Finally, using the fact that $\mathcal{G}_{\mathrm{PTF}}$ η-fools degree-d PTFs over $\{-1,1\}^H$, Equation (5) follows from Fact 16 and this completes the proof. ◀

3.3.1 Composing single atomic steps: Proof of Theorem 15 given Lemma 17

At a very high level, Theorem 15 follows by recursive applications of Lemma 17. Given a degree-d polynomial p, Build-Restrictions begins by calling the subroutine Build-Restrictions-Atomic of Lemma 17, which returns a set $\mathcal{R}_{\mathrm{atomic}}(p) =: \mathcal{R}^{(1)}$ of $\exp(s(m,d,\eta))$ many restrictions satisfying the conclusion of Lemma 17. We call a restriction

$\rho \in \mathcal{R}_{\mathrm{atomic}}(p)$ *good* if p_ρ is either τ-regular or satisfies $\mathbf{Var}[p_\rho] \leq \delta\,\mathbf{E}[p_\rho]^2$ (i.e. if p_ρ satisfies the conclusion of Lemma 17), and we call ρ *bad* otherwise. By Lemma 17, we have that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(1)}}[\,\boldsymbol{\rho} \text{ is good}\,] \geq 2^{-C_2 d}$$

and

$$\left|\mathop{\mathbf{E}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(1)}}\left[\mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p_{\boldsymbol{\rho}}(\boldsymbol{x}))]\right] - \mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p(\boldsymbol{x}))]\right| \leq \eta.$$

Build-Restrictions cycles through all $\rho \in \mathcal{R}_{\mathrm{atomic}}(p)$ and determines if each is good or bad. Note that this can be done deterministically in overall time $|\mathcal{R}_{\mathrm{atomic}}(p)| \cdot \mathrm{poly}(n^d)$ via the Fourier formulas $\mathrm{Inf}_i(q) = \sum_{S\ni i} \widehat{q}(S)^2$ and $\mathbf{Var}(q) = \sum_{S\neq\emptyset} \widehat{q}(S)^2$. For each bad restriction ρ, Build-Restrictions recursively calls the subroutine Build-Restrictions-Atomic on the restricted polynomial p_ρ, obtaining another set $\mathcal{R}_{\mathrm{atomic}}(p_\rho)$ of $\exp(s(m,d,\eta))$ many restrictions satisfying the conclusion of Lemma 17. Consider the overall set of restrictions comprising of the good restrictions in $\mathcal{R}_{\mathrm{atomic}}(p)$, along with the the bad $\rho \in \mathcal{R}_{\mathrm{atomic}}(p)$ extended by those in $\mathcal{R}_{\mathrm{atomic}}(p_\rho)$, i.e.

$$\mathcal{R}^{(2)} := \{\rho\colon \rho \in \mathcal{R}_{\mathrm{atomic}}(p) \text{ is good}\} \cup \{\rho \circ \rho'\colon \rho \in \mathcal{R}_{\mathrm{atomic}}(p) \text{ is bad, } \rho' \in \mathcal{R}_{\mathrm{atomic}}(p_\rho)\}.$$

We have that

$$\begin{aligned}\mathop{\mathbf{Pr}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(2)}}[\,\boldsymbol{\rho} \text{ is good}\,] &= 1 - \mathop{\mathbf{Pr}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(2)}}[\,\boldsymbol{\rho} \text{ is bad}\,] \\ &= 1 - \mathop{\mathbf{Pr}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(1)}}[\,\boldsymbol{\rho} \text{ is bad}\,] \cdot \mathop{\mathbf{Pr}}_{\substack{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(1)}\\ \boldsymbol{\rho}'\leftarrow\mathcal{R}_{\mathrm{atomic}}(p_{\boldsymbol{\rho}})}}[\,\boldsymbol{\rho}' \text{ is bad} \mid \boldsymbol{\rho} \text{ is bad}\,] \\ &\geq 1 - (1 - 2^{-C_2 d})^2,\end{aligned}$$

where C_2 is the universal constant from Lemma 17, and

$$\left|\mathop{\mathbf{E}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(2)}}\left[\mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p_{\boldsymbol{\rho}}(\boldsymbol{x}))]\right] - \mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p(\boldsymbol{x}))]\right| \leq 2\eta.$$

Iterating this argument and defining $\mathcal{R}^{(j)}$ analogously for $j > 2$, we have that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(j)}}[\,\boldsymbol{\rho} \text{ is good}\,] \geq 1 - (1 - 2^{-C_2 d})^j \tag{6}$$

and

$$\left|\mathop{\mathbf{E}}_{\boldsymbol{\rho}\leftarrow\mathcal{R}^{(j)}}\left[\mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p_{\boldsymbol{\rho}}(\boldsymbol{x}))]\right] - \mathop{\mathbf{E}}_{\boldsymbol{x}\leftarrow\{-1,1\}^n}[\mathrm{sign}(p(\boldsymbol{x}))]\right| \leq j\eta. \tag{7}$$

By choosing $j = 2^{O(d)}\log(1/\epsilon)$ we can make the RHS of Equation (6) at least $1 - \epsilon$, and by our choice of $\eta = \epsilon/2^{O(d)}\log(1/\epsilon)$ we have that the RHS of Equation (7) is at most ϵ. Finally, we note that

$$|\mathcal{R}^{(j)}| \leq \exp(s(m,d,\eta)\cdot j) = \exp(s(m,d,\eta)\cdot 2^{O(d)}\log(\tfrac{1}{\epsilon}))$$

and this completes the proof of Theorem 15 given Lemma 17. ◀

4 Instantiating our derandomized regularity lemma: Proofs of Theorems 1 and 2

4.1 The $\mathcal{G}_{\mathrm{PTF}}$ PRG

To apply Theorem 15 we need a PRG $\mathcal{G}_{\mathrm{PTF}}$ that (i) is $4d$-wise uniform, and (ii) η-fools degree-d PTFs over $\{-1,1\}^m$. Since m (the number of variables) is quite small in Theorem 15, the idea is to use a PRG which has a poor dependence on this parameter but a good dependence on the error parameter η and the degree parameter d, since we will be able to take advantages of these good dependences on η and d while not having to "pay too much" for the poor dependence on m.

As mentioned earlier, the PRG for degree-d PTFs from [23] is well suited to the purpose of achieving item (ii) above with good parameters for us. We recall the performance guarantee of [23]:

▶ **Theorem 18** (Special case[3] of Theorem 2 of [23])**.** *There is an efficient explicit PRG with seed length $2^{O(\sqrt{d\log m})} + \mathrm{polylog}(1/\eta)$ that η-fools the class of degree-d PTFs over $\{-1,1\}^m$.*

Regarding item (i), it is not clear that the [23] PRG is $4d$-wise uniform but we can easily augment it to be $4d$-wise uniform by simply performing a bitwise XOR with a $4d$-wise uniform distribution. The correctness of this is ensured by the following simple lemma:

▶ **Lemma 19.** *Let $G_1 : \{-1,1\}^{s_1} \to \{-1,1\}^m$ be a PRG that ϵ-fools the class of degree-d PTFs over $\{-1,1\}^m$. Let $G_2 : \{-1,1\}^{s_2} \to \{-1,1\}^m$ be such that the distribution of $G_2(\boldsymbol{r})$ is k-wise uniform for $\boldsymbol{r}$ uniform random over $\{-1,1\}^{s_2}$. Define $G : \{-1,1\}^{s_1+s_2} \to \{-1,1\}^m$ by*

$$(G(r_1,r_2))_j = (G(r_1))_j \cdot (G(r_2))_j \qquad \textit{for } j \in [m].$$

Then (i) G ϵ-fools the class of degree-d PTFs over $\{-1,1\}^m$, and (ii) $G(\boldsymbol{r}_1,\boldsymbol{r}_2)$ is k-wise uniform for $(\boldsymbol{r}_1,\boldsymbol{r}_2)$ uniform random over $\{-1,1\}^{s_1+s_2}$.

Proof. For (i), observe that if $p(x_1,\dots,x_m)$ is a degree-d polynomial, then for any fixed $a \in \{-1,1\}^m$ the function $q(x_1,\dots,x_m) = p(a_1x_1,\dots,a_mx_m)$ is also a degree-d polynomial. It follows that for any fixed setting of $r_1 \in \{-1,1\}^{s_2}$, the distribution

$$(G(\boldsymbol{r}_1,r_2))_{\boldsymbol{r}_1\leftarrow\{-1,1\}^{s_1}} = ((G(\boldsymbol{r}_1))_1\cdot(G(r_2))_1,\dots,(G(\boldsymbol{r}_1))_m\cdot(G(r_2))_m)_{\boldsymbol{r}_1\leftarrow\{-1,1\}^{s_1}}$$

ϵ-fools the class of degree-d PTFs over $\{-1,1\}^m$, and (i) follows directly from this.

For (ii), similarly observe that if $\boldsymbol{X} = (\boldsymbol{X}_1,\dots,\boldsymbol{X}_m)$ is a k-wise uniform random variable over $\{-1,1\}^m$ then for any fixed $a \in \{-1,1\}^m$, the random variable $(a_1\boldsymbol{X}_1,\dots,a_m\boldsymbol{X}_m)$ is also k-wise uniform. It follows that for any fixed setting of $r_1 \in \{-1,1\}^{s_1}$, the distribution

$$(G(r_1,\boldsymbol{r}_2))_{\boldsymbol{r}_2\leftarrow\{-1,1\}^{s_2}} = ((G(r_1))_1\cdot(G(\boldsymbol{r}_2))_1,\dots,(G(r_1))_m\cdot(G(\boldsymbol{r}_2))_m)_{\boldsymbol{r}_2\leftarrow\{-1,1\}^{s_2}}$$

is k-wise uniform, and (ii) follows directly from this. ◀

[3] Theorem 2 of [23] gives a PRG with seed length $2^{O(\sqrt{\log S})} + \mathrm{polylog}(1/\eta)$ that η-fools the class of size-S Threshold-of-AC^0 circuits. The current statement follows as a special case of this by observing that any degree-d m-variable PTF can be viewed as a Threshold-of-AND circuit of size at most $S = O(m^d)$, since there are at most $\binom{m}{0} + \cdots + \binom{m}{d}$ many ANDs over at most d out of m input variables.

Recalling the well-known fact (see e.g. [24]) that there are simple explicit pseudorandom generators with seed length $O(k \log m)$ that output k-wise independent distributions over $\{-1,1\}^m$, we get the following corollary of Theorem 18:

▶ **Corollary 20.** *There is an efficient explicit PRG $\mathcal{G}_{\mathrm{PTF}}$ with seed length $s(m,d,\eta) = 2^{O(\sqrt{d \log m})} + \mathrm{polylog}(1/\eta)$ that is $4d$-wise uniform and η-fools the class of degree-d PTFs over $\{-1,1\}^m$.*

4.2 Proof of Theorem 1

Recall that to prove Theorem 1, we must give a deterministic algorithm for ϵ-approximate counting n-variable degree-d PTFs over Boolean space that runs in time

$$\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right) \cdot n^{2^{O(d)} \cdot \mathrm{poly}(1/\epsilon)}.$$

The algorithm operates in two stages. In the first stage, it runs the Build-Restrictions procedure given in Theorem 15 with its "ϵ" and "δ" parameters both set to ϵ, its "τ" parameter set to $(\epsilon/d)^{O(d)}$, and $\mathcal{G}_{\mathrm{PTF}}$ being the PRG given in Corollary 20. With these parameter settings the "m" of Theorem 15 is $m = (d/\epsilon)^{O(d)}$ and the "η" is $\eta = \frac{\epsilon}{2^{O(d)} \log(\frac{1}{\epsilon})}$. Recall that the running time of Build-Restrictions is

$$\begin{aligned}
&\mathrm{poly}(n^d, \exp(s(m,d,\eta) \cdot 2^{O(d)} \log(\tfrac{1}{\epsilon}))) \\
&= \mathrm{poly}(n^d, \exp((2^{O(d\sqrt{\log(d/\epsilon)})} + \mathrm{poly}(d, \log(1/\epsilon))) \cdot 2^{O(d)} \log(\tfrac{1}{\epsilon}))) \\
&= \mathrm{poly}(n^d, \exp(2^{O(d\sqrt{\log(d/\epsilon)})})), \qquad (8)
\end{aligned}$$

and that Build-Restrictions outputs a set $\mathcal{R}$ of at most

$$\exp(s(m,d,\eta) \cdot 2^{O(d)} \log(\tfrac{1}{\epsilon})) = \exp(2^{O(d\sqrt{\log(d/\epsilon)})})$$

many restrictions.

In the second stage, the algorithm exhaustively iterates over each restriction $\rho \in \mathcal{R}$. For each $\rho \in \mathcal{R}$ it computes a value v_ρ as follows:

1. First, it computes $\mathrm{Inf}_i(p_\rho)$ for all variables i that were not fixed by the restriction ρ (recall that this can be done deterministically in time $\mathrm{poly}(n^d)$ via the Fourier formula $\mathrm{Inf}_i(p_\rho) = \sum_{S \ni i} \hat{p_\rho}(S)^2$. It uses these computed influences to determine whether or not p_ρ is τ-regular according to Definition 8 (recall that $\tau = (\epsilon/d)^{O(d)}$).
2. If p_ρ is τ-regular, then it runs the [12] deterministic PRG-based algorithm for Gaussian space (recall Table 2) to obtain a $\pm\epsilon$-accurate estimate of $\mathbf{E}_{\boldsymbol{g} \sim N(0,1)^n}[\mathrm{sign}(p_\rho(\boldsymbol{g})]$. (Recall that the [12] algorithm takes time $n^{2^{O(d)} \cdot \mathrm{poly}(1/\epsilon)}$.) It sets v_ρ to be the output of this algorithm.
3. Otherwise, if p_ρ is not τ-regular, it simply sets v_ρ to be the value $\mathrm{sign}(\mathbf{E}[p_\rho]) = \mathrm{sign}(\widehat{p_\rho}(\emptyset)) \in \{-1,1\}$.

The final value v returned by the algorithm is the average over all $\rho \in \mathcal{R}$ of the values v_ρ.

From Equation (8) and item (2) above we have that the running time of the algorithm is

$$\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right) \cdot n^{2^{O(d)} \cdot \mathrm{poly}(1/\epsilon)},$$

as claimed in Theorem 1. To conclude the proof it remains only to argue that v is indeed within an additive $\pm O(\epsilon)$ of the true value of $\mathbf{E}_{\boldsymbol{x} \leftarrow \{-1,1\}^n}[\mathrm{sign}(p(\boldsymbol{x}))]$. Recalling Equation (3) and the fact that at most an ϵ fraction of restrictions $\rho \in \mathcal{R}$ are such that neither is p_ρ τ-regular nor is $\mathrm{sign}(p_\rho)$ δ-close (i.e. ϵ-close) to the constant function $\mathrm{sign}(\mathbf{E}[p_\rho])$, using item (2) above it suffices to show that for every ρ such that p_ρ is τ-regular, it holds that

$$\left| \mathop{\mathbf{E}}_{\boldsymbol{g}\sim N(0,1)}[\mathrm{sign}(p_\rho(\boldsymbol{g})] - \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\{-1,1\}^n}[\mathrm{sign}(p_\rho(\boldsymbol{x})] \right| \leq O(\epsilon).$$

Since p_ρ is τ-regular, we have that

$$\max_{i\in[n]} \mathrm{Inf}_i(p_\rho) \leq \sqrt{\sum_{j=i}^{n} \mathrm{Inf}_i(p_\rho)^2} \leq \tau \sum_{j=1}^{n} \mathrm{Inf}_j(p_\rho) \leq \tau d \cdot \mathbf{Var}[p_\rho],$$

where the last inequality uses that the total influence of a degree-d polynomial is at most d times its variance. Hence by the invariance principle (Theorem 7), we have that

$$\left| \mathop{\mathbf{E}}_{\boldsymbol{g}\sim N(0,1)}[\mathrm{sign}(p_\rho(\boldsymbol{g})] - \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\{-1,1\}^n}[\mathrm{sign}(p_\rho(\boldsymbol{x})] \right| \leq O(d(\tau d)^{1/8d}) = O(\epsilon)$$

as desired, where the last inequality is by our choice of $\tau = (\epsilon/d)^{O(d)}$. This concludes the proof of Theorem 1. ◀

4.3 Proof of Theorem 2

To prove Theorem 2, we must give a deterministic algorithm for ϵ-approximate counting n-variable degree-d PTFs over Boolean space that runs in time $n^{O_{d,\kappa}(1/\epsilon)^\kappa}$.

The first stage of the algorithm here is identical to the first stage of the algorithm in the previous subsection, with the same parameter settings and running time. The second stage differs only in item (2), where now in the τ-regular case we run the [14] deterministic PRG-based algorithm for Gaussian space, which runs in time $n^{O_{d,\kappa}(1/\epsilon)^\kappa}$.

The analysis of correctness (showing that the output of this algorithm is $\pm O(\epsilon)$-close to the right value) is identical to the previous subsection. The running time of the algorithm is $\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right) \cdot n^{O_{d,\kappa}(1/\epsilon)^\kappa}$, where we recall that the function of d and $1/\kappa$ hidden by the big-Oh notation is very fast-growing, in fact of Ackermann type. We now observe that the first component of the running time, $\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right)$, is asymptotically dominated by the second $n^{O_{d,\kappa}(1/\epsilon)^\kappa}$ component, which gives us the final claimed $n^{O_{d,\kappa}(1/\epsilon)^\kappa}$ runtime. We establish this by comparing d and ϵ as follows:

- If d is less than $(\log(1/\epsilon))^{1/3}$, then the first component $\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right)$ is less than $\exp(2^{(\log(1/\epsilon))^{0.9}})$, whereas the second expression is $n^{O_{d,\kappa}(1)\cdot(1/\epsilon)^\kappa} \geq \exp(O_{d,\kappa}(1)\cdot(1/\epsilon)^\kappa)$. For $(1/\epsilon)^\kappa$ to be as small as $2^{(\log(1/\epsilon))^{0.9}}$ we would need $\kappa \leq (\log(1/\epsilon))^{-0.1}$, but having κ be this small means that the $O_{d,\kappa}(1)$ factor will make $O_{d,\kappa}(1)\cdot(1/\epsilon)^\kappa$ much bigger than $2^{(\log(1/\epsilon))^{0.9}}$.
- If d is larger than $(\log(1/\epsilon))^{1/3}$, then the first expression $\exp\left(2^{O(d\sqrt{\log(d/\epsilon)})}\right)$ is less than $\exp(2^{d^3})$, whereas the second expression is still at least $\exp(O_{d,\kappa}(1)\cdot(1/\epsilon)^\kappa)$. Irrespective of the value of κ, the $O_{d,\kappa}(1)$ in this second expression asymptotically dominates $\exp(2^{d^3})$.

This concludes the proof of Theorem 2. ◀

References

1 Noga Alon, Gregory Gutin, and Michael Krivelevich. Algorithms with large domination ratio. *J. Algorithms*, 50(1):118–134, 2004.

2 Aline Bonami. Etude des coefficients Fourier des fonctiones de $L^p(G)$. *Ann. Inst. Fourier (Grenoble)*, 20(2):335–402, 1970.

3 Anindya De, Ilias Diakonikolas, and Rocco A. Servedio. Deterministic approximate counting for juntas of degree-2 polynomial threshold functions. In *Proceedings of the 29th Annual Conference on Computational Complexity (CCC)*, pages 229–240. IEEE, 2014.

4 Anindya De and Rocco A. Servedio. Efficient deterministic approximate counting for low-degree polynomial threshold functions. In *Proceedings of the 46th Annual Symposium on Theory of Computing (STOC)*, pages 832–841, 2014.

5 Ilias Diakonikolas, Daniel M. Kane, and Jelani Nelson. Bounded independence fools degree-2 threshold functions. In *Proc. 51st IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 11–20, 2010.

6 Ilias Diakonikolas, Rocco A. Servedio, Li-Yang Tan, and Andrew Wan. A regularity lemma and low-weight approximators for low-degree polynomial threshold functions. *Theory of Computing*, 10:27–53, 2014.

7 Leonard Gross. Logarithmic Sobolev inequalities. *Amer. J. Math.*, 97(4):1061–1083, 1975.

8 Prahladh Harsha, Adam Klivans, and Raghu Meka. Bounding the sensitivity of polynomial threshold functions. *Theory of Computing*, 10(1):1–26, 2014. URL: `http://www.theoryofcomputing.org/articles/v010a001`.

9 Valentine Kabanets, Daniel M Kane, and Zhenjian Lu. A polynomial restriction lemma with applications. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 615–628, 2017.

10 Valentine Kabanets and Zhenjian Lu. Satisfiability and Derandomization for Small Polynomial Threshold Circuits. In *Proceedings of the 22nd International Conference on Randomization and Computation (RANDOM)*, volume 116, pages 46:1–46:19, 2018.

11 Daniel Kane. k-independent Gaussians fool polynomial threshold functions. In *Proceedings of the 26th Conference on Computational Complexity (CCC)*, pages 252–261, 2011.

12 Daniel Kane. A small PRG for polynomial threshold functions of Gaussians. In *Proceedings of the 52nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 257–266, 2011.

13 Daniel Kane. A structure theorem for poorly anticoncentrated Gaussian chaoses and applications to the study of polynomial threshold functions. In *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 91–100, 2012.

14 Daniel Kane. A pseudorandom generator for polynomial threshold functions of Gaussians with subpolynomial seed length. In *Proceedings of the 29th Annual Conference on Computational Complexity (CCC)*, pages 217–228, 2014.

15 Daniel Kane. A polylogarithmic PRG for degree 2 threshold functions in the Gaussian setting. In *Proceedings of the 30th Conference on Computational Complexity (CCC)*, pages 567–581, 2015.

16 Daniel Kane and Raghu Meka. A PRG for Lipschitz functions of polynomials with applications to sparsest cut. In *Proceedings of the 45th Annual Symposium on Theory of Computing (STOC)*, pages 1–10, 2013.

17 Daniel Kane and Sankeerth Rao. A PRG for Boolean PTF of degree 2 with seed length subpolynomial in ε and logarithmic in n. In *Proceedings of the 33rd Computational Complexity Conference (CCC)*, pages 2:1–2:24, 2018.

18 Daniel M. Kane. The Correct Exponent for the Gotsman-Linial Conjecture. *computational complexity*, 23:151–175, 2014.

19 Raghu Meka and David Zuckerman. Pseudorandom generators for polynomial threshold functions. *SIAM Journal on Computing*, 42(3):1275–1301, 2013.

20 Elchannan Mossel, Ryan O'Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: Invariance and optimality. *Annals of Mathematics*, 171:295–341, 2010.

21 Ryan O'Donnell. *Analysis of Boolean Functions.* Cambridge University Press, 2014. Available at `http://analysisofbooleanfunctions.net/`.

22 Ryan O'Donnell, Rocco A. Servedio, and Li-Yang Tan. Fooling Gaussian PTFs via Local Hyperconcentration. In *Proceedings of the 51st Annual Symposium on Theory of Computing (STOC)*, 2020. to appear.

23 Rocco A. Servedio and Li-Yang Tan. Luby-Velickovic-Wigderson Revisited: Improved Correlation Bounds and Pseudorandom Generators for Depth-Two Circuits. In *Proceedings of the 22nd International Conference on Randomization and Computation (RANDOM)*, pages 56:1–56:20, 2018.

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

A Proof of Fact 6

Let $p : \{-1,1\}^n \to \mathbb{R}$ be a degree-d multilinear polynomial normalized so that $\mathbf{Var}[p]$ (which is equal to $\sum_{\emptyset \neq S, |S| \leq d} \widehat{p}(S)^2$) equals 1, and let $\mathcal{D}$ be a $4d$-wise uniform distribution over $\{-1,1\}^n$. We consider the mean-zero random variable $p(\boldsymbol{z}) - \mathbf{E}[p(z)] = p(\boldsymbol{z}) - \widehat{p}(\emptyset)$, where $\boldsymbol{z} \leftarrow \mathcal{D}$. We have that $\mathbf{E}[p(\boldsymbol{z})] = 0$, $\mathbf{E}[p(\boldsymbol{z})^2] = \mathbf{Var}_{\boldsymbol{z}}[p(\boldsymbol{z})] = \mathbf{Var}_{\boldsymbol{x} \leftarrow \{-1,1\}^n}[p(\boldsymbol{x})] = 1$, and by Corollary 4 (derandomized (2,4)-Hypercontractivity), we further have that $\mathbf{E}[p(\boldsymbol{z})^4] \leq 9^d\, \mathbf{E}[p(\boldsymbol{z})^2]^2 = 9^d$. Fact 6 now follows from the following simple fact:

▶ **Fact 21** ([1], Lemma 3.2). *Let $\boldsymbol{A}$ be a real valued random variable satisfying $\mathbf{E}[\boldsymbol{A}] = 0, \mathbf{E}[\boldsymbol{A}^2] = 1$ and $\mathbf{E}[\boldsymbol{A}^4] \leq b$. Then $\mathbf{Pr}[\boldsymbol{A} \geq 1/(4\sqrt{b})] \geq 1/(4^{4/3}b)$.*

B Proof of Lemma 13 and Lemma 14

B.1 Proof of Lemma 13 (derandomized Lemma 5.2 of [8]: large critical index)

We express $p(x)$ as $q(x) + r(x) + \mathbf{E}[p]$, where

$$q(x) = \sum_{S \not\subseteq K} \hat{p}(S)\chi_S(x) \quad \text{and} \quad r(x) = \sum_{\substack{S \subseteq K \\ S \neq \emptyset}} \hat{p}(S)\chi_S(x).$$

[8]'s Lemma 5.2 follows from the following two claims:

1. For every constant c there is a sufficiently large constant C_1 such that if $K := 2^{C_1 d} \log(1/\delta)/\tau^2$, then

$$\mathop{\mathbf{E}}_{\boldsymbol{\rho} \leftarrow \{-1,1\}^K}[\mathbf{Var}(p_{\boldsymbol{\rho}})] \leq \delta \cdot 2^{-cd} \cdot \mathop{\mathbf{E}}_{\boldsymbol{\rho} \leftarrow \{-1,1\}^K}[r(\boldsymbol{\rho})^2]. \tag{9}$$

(This is [8]'s Claim 5.6.) Consequently, by Markov's inequality, for all constants c and c' we can again choose C_1 to be sufficiently large to ensure that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho} \leftarrow \{-1,1\}^K}\left[\mathbf{Var}(p_{\boldsymbol{\rho}}) \leq \delta \cdot 2^{-cd} \mathop{\mathbf{E}}_{\boldsymbol{\rho} \leftarrow \{-1,1\}^K}[r(\boldsymbol{\rho})^2]\right] \geq 1 - 2^{-c'd}. \tag{10}$$

2. There are constants b and b' such that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{\rho} \leftarrow \{-1,1\}^K}\left[\mathbf{E}[p_{\boldsymbol{\rho}}]^2 \geq 2^{-bd} \mathop{\mathbf{E}}_{\boldsymbol{\rho} \leftarrow \{-1,1\}^K}[r(\boldsymbol{\rho})^2]\right] \geq 2^{-b'd}, \tag{11}$$

which follows from an application of Lemma 5.4 of [8].

The proof of Lemma 13 follows [8]'s proof of their Lemma 5.2 almost exactly. The only changes are that:

- the functions $\rho \mapsto \mathbf{Var}(p_\rho)$ and $\rho \mapsto r(\rho)^2$ are degree $2d$ polynomials in ρ, and hence Equations (9) and (10) both hold for $\boldsymbol{\rho}$ drawn from any $2d$-wise independent distribution over $\{-1,1\}^K$;
- we use our derandomized version of Lemma 5.4 of [8], namely Fact 6, in place of Lemma 5.4 of [8] to deduce that Equation (11) also holds for $\boldsymbol{\rho}$ drawn from any $2d$-wise independent distribution over $\{-1,1\}^K$.

The rest of the proof is unchanged.

B.2 Proof of Lemma 14 (derandomized Lemma 5.1 of [8]: small critical index)

The proof of Lemma 14 follows [8]'s proof of their Lemma 5.1 almost exactly. The only changes are that

- we use our derandomized version of (2,4)-Hypercontractivity, namely Corollary 4, in place of (2,4)-Hypercontractivity (Lemma 4.3 of [8], which is used in the line immediately following Equation (5.1) of their paper);
- we use our derandomized version of Lemma 5.4 of [8], namely Fact 6, in place of Lemma 5.4 of [8], which is used two lines after Equation (5.2) of their paper.

The rest of the proof is unchanged.

▶ Remark 22 (Motivating our notion of regularity). Recall from Section 1.4 that the works of [6, 18] use a technically slightly different notion of "regularity." In those works an n-variable multivariate polynomial p is considered to be τ-regular if for every $i \in [n]$ we have that $\mathrm{Inf}_i(p) \le \tau \cdot \sum_{i=1}^n \mathrm{Inf}_i(p)$. Intuitively, we may view this as a notion of "regularity-in-ℓ_∞", and the notion used in the current paper and in [8, 19] as a notion of "regularity-in-ℓ_2".

We remark here that the small critical index case (the subject of Lemma 14 and of Lemma 5.1 of [8]) is the reason why we need to work with the [8] notion of regularity-in-ℓ_2 given in Definition 8 rather than the regularity-in-ℓ_∞ notion used in [6, 18]. In more detail, to handle the small critical index case using the regularity-in-ℓ_∞ notion, the analysis of [6, 18] uses an exponential tail bound for degree-d polynomials (the "degree-d Chernoff bound", see Theorem 9.23 of [21]). However, derandomizing this degree-d Chernoff bound requires dq-wise uniform distributions, where q depends on the parameters with which the degree-d Chernoff bound is being applied, and it turns out that because of the way that the [6, 18] arguments employ the degree-d Chernoff bound, this can be prohibitively expensive in our context. In contrast, recall from Section 2.1 that derandomizing Theorem 3 and Fact 5 (which is all that is needed to establish a derandomized version of the small critical index case using the regularity-in-ℓ_2 notion, as outlined above) can be done using only $4d$-wise uniformity.