Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 145 (2017)

Quantified Derandomization of Linear Threshold Circuits

Roei Tell *

November 6, 2017

Abstract

One of the prominent current challenges in complexity theory is the attempt to prove lower bounds for $\mathcal{TC}^0$, the class of constant-depth, polynomial-size circuits with majority gates. Relying on the results of Williams (2013), an appealing approach to prove such lower bounds is to construct a non-trivial derandomization algorithm for $\mathcal{TC}^0$. In this work we take a first step towards the latter goal, by proving the first positive results regarding the derandomization of $\mathcal{TC}^0$ circuits of depth $d > 2$.

Our first main result is a *quantified derandomization* algorithm for $\mathcal{TC}^0$ circuits with a super-linear number of wires. Specifically, we construct an algorithm that gets as input a $\mathcal{TC}^0$ circuit C over n input bits with depth d and $n^{1+\exp(-d)}$ wires, runs in almost-polynomial-time, and distinguishes between the case that C rejects at most $2^{n^{1-1/5d}}$ inputs and the case that C accepts at most $2^{n^{1-1/5d}}$ inputs. In fact, our algorithm works even when the circuit C is a linear threshold circuit, rather than just a $\mathcal{TC}^0$ circuit (i.e., C is a circuit with linear threshold gates, which are stronger than majority gates).

Our second main result is that even a *modest improvement* of our quantified derandomization algorithm would yield a non-trivial algorithm for *standard derandomization* of all of $\mathcal{TC}^0$, and would consequently imply that $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$. Specifically, if there exists a quantified derandomization algorithm that gets as input a $\mathcal{TC}^0$ circuit with depth d and $n^{1+O(1/d)}$ wires (rather than $n^{1+\exp(-d)}$ wires), runs in time at most $2^{n^{\exp(-d)}}$, and distinguishes between the case that C rejects at most $2^{n^{1-1/5d}}$ inputs and the case that C accepts at most $2^{n^{1-1/5d}}$ inputs, then there exists an algorithm with running time $2^{n^{1-\Omega(1)}}$ for *standard derandomization* of $\mathcal{TC}^0$.

*Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, Israel. Email: roei.tell@weizmann.ac.il

ISSN 1433-8092

Contents

1 Introduction

The classical problem of *derandomization of a circuit class* $\mathcal{C}$ is the following: Given a circuit $C \in \mathcal{C}$, deterministically distinguish between the case that the acceptance probability of C is at least $2/3$ and the case that the acceptance probability of C is at most $1/3$. When $\mathcal{C} = \mathcal{P}/\text{poly}$, this problem can be solved in polynomial time if and only if *promise-*$\mathcal{BPP}$ $=$ *promise-*$\mathcal{P}$. However, at the moment we do not know how to solve the problem in polynomial time even if $\mathcal{C}$ is the class of polynomial-sized CNFs.

The derandomization problem for a circuit class $\mathcal{C}$ is tightly related to lower bounds for $\mathcal{C}$. Relying on the classic hardness-randomness paradigm [Yao82, BM84, NW94], sufficiently strong lower bounds for a class $\mathcal{C}$ imply the existence of pseudorandom generators with short seed for $\mathcal{C}$, which allow to derandomize $\mathcal{C}$ (see, e.g., [AB09, Chp. 20], [Gol08, Chp. 8.3]). On the other hand, the existence of a non-trivial derandomization algorithm for a circuit class $\mathcal{C}$ typically implies (weak) lower bounds for $\mathcal{C}$. Specifically, for many specific classes $\mathcal{C}$ (e.g., $\mathcal{C} = \mathcal{P}/\text{poly}$), the existence of a derandomization algorithm for $\mathcal{C}$ running in time $2^n/n^{\omega(1)}$ implies that $\mathcal{E}^{\mathcal{NP}} \not\subseteq \mathcal{C}$, and in some cases also that $\mathcal{NEXP} \not\subseteq \mathcal{C}$ (see [Wil13, SW13, BV14], which build on [IW98, IKW02]).

Following Williams' proof that $\mathcal{ACC}$ does not contain $\mathcal{NEXP}$ [Wil11], one of the prominent current challenges in complexity theory is the attempt to prove similar lower bounds for the complexity class $\mathcal{TC}^0$ (i.e., the class of constant-depth, polynomial-sized circuits with majority gates, which extends $\mathcal{ACC}$). Even after extensive efforts during the last few decades (and with renewed vigor recently), the best-known lower bounds for $\mathcal{TC}^0$ assert the existence of functions in $\mathcal{P}$ that require $\mathcal{TC}^0$ circuits with a *slightly super-linear* number of wires, or with a *linear* number of gates (see Section 2 for further background).

Since derandomization algorithms imply lower bounds in general, an appealing approach to prove lower bounds for $\mathcal{TC}^0$ is to construct derandomization algorithms for this class. Moreover, a non-trivial derandomization of $\mathcal{TC}^0$ would separate $\mathcal{TC}^0$ from $\mathcal{NEXP}$ (and not only from $\mathcal{E}^{\mathcal{NP}}$; see [SW13, BV14]). Accordingly, the problem of either derandomizing $\mathcal{TC}^0$ or constructing a deterministic algorithm for *satisfiability* of $\mathcal{TC}^0$ (which would be a stronger result) was recently suggested as a central open problem in complexity theory both by Williams [Wil14a] and by Aaronson [Aar17]. [1]

An intensive recent effort has been devoted to constructing deterministic algorithms for satisfiability $\mathcal{TC}^0$. Such algorithms (with non-trivial running time) have been constructed for $\mathcal{TC}^0$ circuits of depth two, and for certain "structured subclasses" of $\mathcal{TC}^0$ (see [IPS13, Wil14b, AS15, SSTT16, Tam16]). However, much less is known about *derandomization* algorithms for $\mathcal{TC}^0$. Following an intensive effort to construct pseudorandom generators for a single *linear threshold function* [DGJ+10, RS10, GOWZ10, KRS12, MZ13, Kan11, Kan14, KM15, GKM15] (i.e., a single "gate"; for background see Sections 2.2 and 4.2), a first step towards derandomizing $\mathcal{TC}^0$ *circuits* was very recently undertaken by Servedio and Tan [ST17b], who considered the problem of derandomizing $\mathcal{TC}^0$ circuits of *depth two*. [2]

In this work we take a significant additional step towards the derandomization of $\mathcal{TC}^0$, by proving the *first positive results* regarding the derandomization of $\mathcal{TC}^0$ circuits of *any constant depth* $d \geq 2$. Loosely speaking, we first construct an algorithm for a "relaxed" type of derandomization problem of sparse $\mathcal{TC}^0$ circuits of any constant depth $d \geq 2$. As far as we are aware of, this is the first deterministic circuit-analysis algorithm for $\mathcal{TC}^0$ circuits of

[1] See the first open problem in the Conclusions section in [Aar17], and Section 4.2 in [Wil14a].

[2] Their manuscript is still unpublished, and so we describe their results in Section 2.2.

any constant depth that do not admit any special structure (other than being sparse). Then, we show that even a modest improvement in the parameters of the foregoing algorithm (for the "relaxed" problem) would yield a non-trivial algorithm for *standard* derandomization of *all of* $\mathcal{TC}^0$; indeed, as mentioned above, such a result would imply that $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$. We thus suggest this approach (of the "relaxed" derandomization problem) as a potentially tractable line-of-attack towards proving $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$ (see Section 1.1.3).

1.1 Our results

Our two main results lie within the framework of *quantified derandomization*. Quantified derandomization, which was introduced by Goldreich and Wigderson [GW14], is the relaxed derandomization problem of distinguishing between a circuit that accepts $1 - o(1)$ of its inputs and a circuit that rejects $1 - o(1)$ of its inputs (where the $1 - o(1)$ term replaces the original $2/3$ term in standard derandomization).

On the one hand, this relaxation potentially allows to construct more efficient derandomization algorithms. But on the other hand, the standard derandomization problem can be *reduced to quantified derandomization*, by applying strong error-reduction within the relevant circuit class (such that a circuit with acceptance probability $2/3$ is transformed to a circuit with acceptance probability $1 - o(1)$). Of course, a main goal underlying this approach is to reduce standard derandomization to a parameter setting for which we are able to construct a corresponding algorithm for quantified derandomization.

1.1.1 A quantified derandomization algorithm

Our first result is a *quantified derandomization algorithm* for $\mathcal{TC}^0$ circuits with a slightly super-linear number of wires. In fact, our algorithm works not only for $\mathcal{TC}^0$, but also for the class of linear threshold circuits: While in $\mathcal{TC}^0$ circuits each gate computes the majority function, in linear threshold circuits each gate computes a linear threshold function (i.e., a function of the form $g(x) = \text{sgn}\left(\sum_{i\in[n]} w_i \cdot x_i - \theta\right)$, for $w \in \mathbb{R}^n$ and $\theta \in \mathbb{R}$; see Section 4.2 for definitions). Towards stating this first result, denote by $\mathcal{C}_{n,d,w}$ the class of linear threshold circuits over n input bits of depth d and with at most w wires.

Theorem 1.1 *(quantified derandomization of linear threshold circuits). There exists a deterministic algorithm that, when given as input a circuit $C \in \mathcal{C}_{n,d,n^{1+2^{-10d}}}$, runs in time $n^{O(\log\log(n))^2}$, and satisfies the following:*

1. *If C accepts all but at most $B(n) = 2^{n^{1-1/5d}}$ of its inputs, then the algorithm accepts C.*
2. *If C rejects all but at most $B(n) = 2^{n^{1-1/5d}}$ of its inputs, then the algorithm rejects C.*

Observe that as d grows larger, the algorithm in Theorem 1.1 solves a more difficult derandomization task (since $B(n)$ is larger), but only has to handle circuits with fewer wires (i.e., $n^{1+\exp(-d)}$). Also note that the algorithm in Theorem 1.1 is "*whitebox*": That is, the algorithm gets as input an *explicit description* of a *specific* linear threshold circuit C, and uses this description when estimating the acceptance probability of C. [3] The actual algorithm that we construct works for a more general parameter regime, which exhibits

[3] The algorithm in Theorem 1.1 works in any reasonable model of explicitly representing linear threshold circuits; see Section 4.2 for a brief discussion.

a trade-off between the number $B(n) = 2^{n^{1-\delta}}$ of exceptional inputs for C and the number $n^{1+\delta \cdot \exp(-d)}$ of wires of C (see Theorem 5.1 for a precise statement).

The limitation on the number of wires of C in Theorem 1.1 (i.e., $n^{1+\exp(-d)}$) essentially *matches the best-known lower bounds for linear threshold circuits*. This is no coincidence: Our algorithm construction follows a common theme in the design of circuit-analysis algorithms (e.g., derandomization algorithms or algorithms for satisfiability), which is the conversion of techniques that underlie lower bound proofs into algorithmic techniques. In this case, we observe that certain proof techniques for *correlation bounds* for a circuit class $\mathcal{C}$ can be used to obtain algorithmic techniques for *quantified derandomization* of $\mathcal{C}$. In particular, to construct the algorithm in Theorem 1.1, we leverage the techniques underlying the recent proof of Chen, Santhanam, and Srinivasan [CSS16] of correlation bounds for linear threshold circuits. A high-level description of our algorithm appears in Section 3.1.

1.1.2 A reduction of standard derandomization to quantified derandomization

Our second result reduces the *standard* derandomization problem of $\mathcal{TC}^0$ to the *quantified* derandomization problem of $\mathcal{TC}^0$ circuits with a *super-linear number* of wires. In fact, we show that even a modest improvement of Theorem 1.1 would yield a non-trivial algorithm for *standard* derandomization of *all of* $\mathcal{TC}^0$.

Theorem 1.2 *(a reduction of standard derandomization to quantified derandomization). Assume that there exists a deterministic algorithm that, when given as input a circuit $C \in \mathcal{C}_{n,d,n^{1+O(1/d)}}$, runs in time at most $T(n) = 2^{n^{1/4^d}}$, and for the parameter $B(n) = 2^{n^{1-1/5d}}$ satisfies the following: If C accepts all but at most $B(n)$ of its inputs then the algorithm accepts C, and if C rejects all but at most $B(n)$ of its inputs then the algorithm rejects C.*

Then, there exists an algorithm that for every $k \in \mathbb{N}$ and $d \in \mathbb{N}$, when given as input a circuit $C \in \mathcal{C}_{m,d,m^k}$, runs in time $2^{m^{1-\Omega(1)}}$, and satisfies the following: If C accepts at least $2/3$ of its inputs then the algorithm accepts C, and if C rejects at least $2/3$ of its inputs then the algorithm rejects C.

The gap between the algorithm constructed in Theorem 1.1 and the algorithm assumed in the hypothesis of Theorem 1.2 is quantitatively very small: Specifically, the algorithm in Theorem 1.1 works when the number of wires in the input circuit C is $n^{1+\exp(-d)}$, whereas the algorithm in the hypothesis of Theorem 1.2 is required to work when the number of wires is $n^{1+O(1/d)}$. Moreover, Theorem 1.2 holds even if this improvement (in the number of wires) comes at the expense of a longer running time; specifically, the conclusion of Theorem 1.2 holds even if the algorithm runs in (sufficiently small) sub-exponential time.

As mentioned in the beginning of Section 1, a non-trivial derandomization of $\mathcal{TC}^0$ implies lower bounds for this class. Specifically, combining Theorem 1.2 with [SW13, Thm 1.5] (see also [BV14]), we obtain the following corollary:

Corollary 1.3 *(quantified derandomization implies lower bounds for $\mathcal{TC}^0$). Assume that there exists a deterministic algorithm as in the hypothesis of Theorem 1.2. Then, $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$.*

The result that we actually prove is stronger and more general than the one stated in Theorem 1.2 (see Theorem 6.10). First, the result holds even if we limit ourselves only to the class $\mathcal{TC}^0$, rather than to the class of linear threshold circuits (i.e., if we interpret the class $\mathcal{C}_{n,d,w}$ as the class of $\mathcal{TC}^0$ circuits over n inputs of depth d and with w wires). And secondly, the hypothesis of the theorem can be modified via a trade-off between the number of exceptional inputs for the circuit C and the number of wires in C.

The proof of Theorem 1.2 is based on developing a very efficient method for error-reduction within sparse $\mathcal{TC}^0$. Specifically, we construct a seeded extractor such that there exists a $\mathcal{TC}^0$ circuit that gets input $x \in \{0,1\}^n$ and computes the outputs of the extractor on x and on all seeds using only a super-linear number of wires (i.e., a circuit of depth d uses $n^{1+O(1/d)}$ wires); as far as we know, this is the first construction of a seeded extractor that is specific to $\mathcal{TC}^0$. This construction extends the study of randomness extraction in weak computational models, which has so far focused on $\mathcal{AC}^0$, on $\mathcal{AC}^0[\oplus]$, and on streaming algorithms [BYRST02, Vio05, Hea08, GVW15, CL16]. The construction is described in high-level in Section 3.2, and a precise statement appears in Proposition 6.9.

1.1.3 Restrictions for sparse $\mathcal{TC}^0$ circuits: A potential path towards $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$

Recall that the best-known lower bounds for $\mathcal{TC}^0$ circuits of arbitrary constant depth d are for circuits with $n^{1+\exp(-d)}$ wires. Our results imply that a certain type of analysis of $\mathcal{TC}^0$ circuits *with only* $n^{1+O(1/d)}$ *wires*, which is common when proving correlation bounds (i.e., average-case lower bounds), might suffice to deduce a lower bound for *all of* $\mathcal{TC}^0$.

Specifically, a common technique to prove correlation bounds for a circuit C is the "restriction method", which (loosely speaking) consists of proving the existence of certain subsets of the domain on which C "simplifies" (i.e., C agrees with a simpler function on the subset). We pose the following open problem: Construct a deterministic algorithm that gets as input a $\mathcal{TC}^0$ circuit C with $n^{1+O(1/d)}$ wires, runs in sufficiently small sub-exponential time, and finds a subset S of size larger than $2^{n^{1-1/5d}}$ such that the acceptance probability of $C\restriction_S$ can be approximated in sufficiently small sub-exponential time (see Open Problem 1 in Section 8 for a precise statement). In Section 8 we show that a resolution of the foregoing problem would imply that $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$; this follows from Theorem 1.2 and from the techniques that underlie the proof of Theorem 1.1.

1.1.4 The special case of depth-2 circuits

In addition to our main results, we also construct an alternative quantified derandomization algorithm for the special case of linear threshold circuits of *depth two*. Specifically, we construct a pseudorandom generator with seed length $\tilde{O}(\log(n))$ for the class of depth-2 linear threshold circuits with $n^{3/2-\Omega(1)}$ wires that either accept all but $B(n) = 2^{n^{\Omega(1)}}$ of their inputs or reject all but $B(n)$ of their inputs. This result is not a corollary of Theorem 1.1, and is incomparable to the pseudorandom generator of Servedio and Tan [ST17b].

The precise result statement and proof appear in Section 7. The generator construction is obtained by leveraging the techniques of Kane and Williams [KW16] for correlation bounds for linear threshold circuits of depth two.

1.2 Organization

In Section 2 we provide background and discuss some relevant previous works. In Section 3 we give high-level overviews of the proofs of Theorems 1.1 and 1.2. After presenting preliminary formal definitions in Section 4, we prove Theorem 1.1 in Section 5 and Theorem 1.2 in Section 6. In Section 7 we construct the pseudorandom generator mentioned in Section 1.1.4. Finally, in Section 8 we formally pose the open problem that was mentioned in Section 1.1.3 and show the consequences of a solution to the problem.

2 Background and previous work

2.1 Lower bounds for linear threshold circuits

The best-known lower bounds for computing explicit functions by linear threshold circuits of a *fixed small depth* have been recently proved by Kane and Williams [KW16]. Specifically, they showed that any depth-two linear threshold circuit computing Andreev's function requires $\tilde{\Omega}(n^{3/2})$ gates and $\tilde{\Omega}(n^{5/2})$ wires. They also showed correlation bounds (i.e,. average-case lower bounds with respect to the uniform distribution) for such circuits with Andreev's function. Extending their worst-case lower bounds to depth three, they proved that any depth-3 circuit with a *top majority gate* that computes a specific polynomial-time computable function also requires $\tilde{\Omega}(n^{3/2})$ gates and $\tilde{\Omega}(n^{5/2})$ wires (the "hard" function is a modification of Andreev's function).

For linear threshold circuits of arbitrary constant depth $d \geq 2$, the best-known lower bounds on the number of wires required to compute explicit functions are only slightly super-linear. Specifically, Impagliazzo, Paturi, and Saks [IPS97] proved that any linear threshold circuit of depth d requires at least $n^{1+\exp(-d)}$ wires to compute the parity function; Chen, Santhanam, and Srinivasan [CSS16] strengthened this by showing correlation bounds for such circuits with parity (as well as with the generalized Andreev function). These lower bounds for parity are essentially tight, since Beame, Brisson, and Ladner [BBL92] (and later [PS94]) constructed a linear threshold circuit with $n^{1+\exp(-d)}$ wires that computes parity. We also mention that linear lower bounds on the number of linear threshold *gates* required to compute explicit functions (e.g., the inner-product function) have been proved in several works during the early '90s, and these gate lower bounds apply even for circuits of unrestricted depth (see [Smo90, GT91, ROS94, Nis93]).

2.2 Derandomization of LTFs and of functions of LTFs

There has been an intensive effort in the last decade to construct pseudorandom generators for a *single linear threshold function*. This problem was first considered by Diakonikolas *et al.* [DGJ+10] (see also [RS10]), and the current state-of-the-art, following [GOWZ10, Kan11, KRS12, MZ13, Kan14, KM15], is the pseudorandom generator of Gopalan, Kane, and Meka [GKM15], which ϵ-fools any LTF with n input bits using a seed of length $\tilde{O}(\log(n/\epsilon))$. Harsha, Klivans, and Meka [HKM12] considered a *conjunction* of linear threshold functions, and constructed a pseudorandom generator for a subclass of such functions (i.e., for a conjunction of *regular* LTFs; see Section 4.2 for a definition). Gopalan *et al.* [GOWZ10] constructed pseudorandom generators for small decision trees in which the leaves are linear threshold functions.

Very recently, Servedio and Tan [ST17b] considered the problem of derandomizing *linear threshold circuits*. For every $\epsilon > 0$, they constructed a pseudorandom generator that $1/\mathrm{poly}(n)$-fools any *depth-2 linear threshold circuit with at most* $n^{2-\epsilon}$ *wires*, using a seed of length $n^{1-\delta}$, where $\delta = \delta_\epsilon > 0$ is a small constant that depends on ϵ. This yields a derandomization of depth-2 linear threshold circuits with $n^{2-\epsilon}$ wires in time $2^{n^{1-\Omega(1)}}$.

2.3 Quantified derandomization

The quantified derandomization problem, which was introduced by Goldreich and Wigderson [GW14], is a generalization of the standard derandomization problem. For a circuit

class $\mathcal{C}$ and a parameter $B = B(n)$, the $(\mathcal{C}, B)$-derandomization problem is the following: Given a description of a circuit $C \in \mathcal{C}$ over n input bits, deterministically distinguish between the case that C accepts all but $B(n)$ of its inputs and the case that C rejects all but $B(n)$ of its inputs. Indeed, the standard derandomization problem is represented by the parameter value $B(n) = \frac{1}{3} \cdot 2^n$. Similarly to standard derandomization, a solution for the quantified derandomization problem of a class $\mathcal{C}$ via a "black-box" algorithm (e.g., via a pseudorandom generator) yields a corresponding lower bound for $\mathcal{C}$ (see Appendix A).

Prior to this work, quantified derandomization algorithms have been constructed for $\mathcal{AC}^0$, for subclasses of $\mathcal{AC}^0[\oplus]$, for polynomials over $\mathbb{F}_2$ that vanish rarely, and for a subclass of $\mathcal{MA}$. On the other hand, reductions of standard derandomization to quantified derandomization are known for $\mathcal{AC}^0$, for $\mathcal{AC}^0[\oplus]$, for polynomials over large finite fields, and for the class $\mathcal{AM}$ (both the algorithms and the reductions appear in [GW14, Tel17]). In some cases, most notably for $\mathcal{AC}^0$, the parameters of the known quantified derandomization algorithms are very close to the parameters of quantified derandomization to which standard derandomization can be reduced (see [Tel17, Thms 1 & 2]).

3 Overviews of the proofs

3.1 A quantified derandomization algorithm for linear threshold circuits

The high-level strategy of the quantified derandomization algorithm is as follows. Given a circuit $C : \{-1,1\}^n \to \{-1,1\}$, the algorithm deterministically finds a set $S \subseteq \{-1,1\}^n$ of size $|S| \gg B(n)$ on which the circuit C simplifies; that is, C agrees with a function from some "simple" class of functions on almost all points in S. If C accepts all but $B(n)$ of its inputs, then the acceptance probability of $C\restriction_S$ will be very high, and similarly, if C rejects all but $B(n)$ of its inputs, then the acceptance probability of $C\restriction_S$ will be very low. The algorithm then distinguishes between the two cases, by enumerating the seeds of a pseudorandom generator for the "simple" class of functions.

Our starting point in order to construct a deterministic algorithm that finds a suitable set S is the recent proof of correlation bounds for sparse linear threshold circuits by Chen, Santhanam, and Srinivasan [CSS16]. Their proof is based on a *randomized* "whitebox" algorithm that gets as input a linear threshold circuit with depth d and $n^{1+\epsilon}$ wires, and restricts all but $n^{1-\epsilon \cdot \exp(d)}$ of the variables such that the restricted circuit can be approximated by a single linear threshold function. Thus, if we are able to modify their algorithm to a deterministic one, we will obtain a quantified derandomization algorithm with the parameters asserted in Theorem 1.1 (i.e., if $\epsilon = \exp(-d)$, then $B(n) \approx |S|/10 > 2^{n^{1-1/5d}}$). [4]

Converting the randomized restriction algorithm into a deterministic algorithm poses several challenges, which will be our focus in this overview. Let us first describe the original algorithm, in high-level. The algorithm iteratively reduces the depth of the circuit. In each iteration it applies a random restriction that keeps every variable alive with probability $p = n^{-\Omega(1)}$, and otherwise assigns a random value to the variable. The main structural lemma of [CSS16] asserts that such a random restriction turns any LTF to be very biased (i.e., $\exp(-n^{\Omega(1)})$-close to a constant function), with probability $1 - n^{-\Omega(1)}$. Hence, after

[4] This approach follows the well-known theme of "leveraging" techniques from lower bound proofs to algorithmic techniques, and in particular to techniques for constructing circuit-analysis algorithms; see, e.g., [LMN93, San10, Bra10, IMZ12, ST12, IMP12, BIS12, GMR13, TX13, CKK+15, ST17b, ST17a]. We also mention that in [CSS16, Sec. 5] their randomized restriction algorithm is used to construct a *randomized* algorithm for *satisfiability* of sparse linear threshold circuits.

applying the restriction, most gates in the bottom layer of the circuit become very biased, and the fan-in of the rest of the gates in the bottom layer significantly decreases (i.e., we expect it to reduce by a factor of $p = n^{-\Omega(1)}$). The algorithm replaces the very biased gates with the corresponding constants, thereby obtaining a circuit that *approximates* the original circuit (i.e., the two circuits agree on all but $2^{-n^{\Omega(1)}}$ of the inputs); and in [CSS16] it is shown that the algorithm can afterwards fix relatively few variables such that the fan-in of each gate that did not become very biased decreases to be at most one (such a gate can be replaced by a variable or a constant). Thus, if the circuit C_i in the beginning of the iteration was of depth i, we obtain a circuit C_{i-1} of depth $i-1$ that approximates C_i.

One obvious challenge in converting the randomized restriction algorithm into a deterministic algorithm is "derandomizing" the main structural lemma; that is, we need to construct a pseudorandom distribution of restrictions that turns any LTF to be very biased, with high probability. The second challenge is more subtle: In each iteration we replace the "current" circuit C_i by a circuit C_{i-1} that agrees with C_i on almost all inputs in the subcube of the n living variables (i.e., the circuits disagree on at most $2^{n-n^{\Omega(1)}}$ inputs). However, in subsequent iterations we will fix almost all of these n variables, such that only $n^{1-\Omega(1)}$ variables will remain alive. Thus, we have no guarantee that C_i and C_{i-1} will remain close after additional restrictions in subsequent iterations; in particular, C_i and C_{i-1} might disagree on *all* of the inputs in the subcube of living variables in the end of the entire process. Of course, this is very unlikely to happen when values for fixed variables are chosen uniformly, but we need to construct a *pseudorandom* distribution of restrictions such that the approximation of each C_i by C_{i-1} is likely to be maintained throughout the process.

3.1.1 Derandomizing the main structural lemma of [CSS16].

Let $\Phi = (w, \theta)$ be an LTF over n input bits, and consider a random restriction ρ that keeps each variable alive with probability $p = n^{-\Omega(1)}$. Peres' theorem implies that the expected distance of $\Phi\restriction_\rho$ from a constant function is approximately $\sqrt{p}$ (see, e.g., [O'D14, Sec. 5.5]). [5] A natural question is whether we can prove a concentration of measure for this distribution. As an illustrative example, consider the majority function $MAJ(x) = \mathrm{sgn}(\sum_{i\in[n]} x_i)$; for any $t \geq 1$, with probability roughly $1 - t \cdot \sqrt{p}$ it holds that $MAJ\restriction_\rho$ is $\exp(-t^2)$-close to a constant function (see Fact 5.3). The main structural lemma in [CSS16] asserts that a similar statement indeed holds for *any LTF* Φ; specifically, they showed that with probability at least $1 - p^{\Omega(1)}$ it holds that $\Phi\restriction_\rho$ is $\exp(-p^{-\Omega(1)})$-close to a constant function.

We construct a distribution over restrictions that can be efficiently sampled using $\tilde{O}(\log(n))$ random bits such that for any LTF Φ and any $t \geq p^{-1/8}$, with probability at least $1 - \tilde{O}(t^2) \cdot \sqrt{p}$ it holds that $\Phi\restriction_\rho$ is $\exp(-t^2)$-close to a constant function. (The actual statement that we prove is more general; see Proposition 5.8 for precise details.) Indeed, this is both an "almost-full derandomization" of the lemma of [CSS16] as well as a refinement of the quantitative bound in the lemma.

The original proof of [CSS16] relies on a technical case analysis that is reminiscent of other proofs that concern LTFs, and is based on the notion of a *critical index* of a vector $w \in \mathbb{R}^n$ (they refer to the ideas underlying such analyses as "the structural theory of linear threshold functions"; see, e.g., [Ser07, DGJ+10], and Definitions 4.3 and 4.4). In each case, the main technical tools that are used are concentration and anti-concentration theorems

[5] Peres' theorem is usually phrased in terms of the noise sensitivity of Φ, but the latter is propotional to its expected bias under a random restriction; for further details see [CSS16, Prop. 8].

for random weighted sums (i.e., Hoeffding's inequality and the Berry-Esséen theorem, respectively), which are used to bound the probability that several specific random weighted sums that are related to the restricted function $\Phi\restriction_\rho$ fall in certain intervals.

To derandomize the original proof, an initial useful observation is the following. We say that a distribution $\mathbf{z}$ over $\{-1,1\}^n$ is ϵ-pseudorandomly concentrated if for any $w \in \mathbb{R}^n$ and any interval $J \subseteq \mathbb{R}$, the probability that $\langle w, \mathbf{z}\rangle$ falls in J is ϵ-close to the probability that $\langle w, \mathbf{u}_n\rangle$ falls in J (where $\mathbf{u}_n$ is the uniform distribution over $\{-1,1\}^n$). In particular, the Berry-Esséen theorem and Hoeffding's inequality approximately hold for pseudorandom sums $\langle w, \mathbf{z}\rangle$ when $\mathbf{z}$ is pseudorandomly concentrated. The observation is that being ϵ-pseudorandomly concentrated is essentially equivalent to being ϵ-pseudorandom for LTFs (see Claim 4.11). [6] In particular, if a distribution $\mathbf{z}$ over $\{-1,1\}^n$ is chosen using the pseudorandom generator of Gopalan, Kane, and Meka [GKM15] for LTFs, which has seed length $\tilde{O}(\log(n/\epsilon))$, then $\mathbf{z}$ is ϵ-pseudorandomly concentrated.

The main part in the proof of the derandomized lemma is a (non-trivial) modification of the original case analysis, in order to obtain an analysis in which all claims hold under a suitably-chosen pseudorandom distribution of restrictions. Since this part of the proof is quite technical and low-level, we defer its detailed description to Section 5.1.1. However, let us mention that our pseudorandom distribution itself is relatively simple: We first choose the variables to keep alive such that each variable is kept alive with probability approximately $p = n^{-\Omega(1)}$, and the choices are $O(1)$-wise independent; and then we independently choose values for the fixed variables, using the generator of [GKM15] with error parameter $\epsilon = 1/\mathrm{poly}(n)$. We also note that it is suprising that in our setting the case analysis can be modified in order to obtain an "almost-full derandomization" (i.e., seed length $\tilde{O}(\log(n))$), since previous derandomizations of similar case analyses regarding LTFs for different settings required much larger seed for error $\epsilon = n^{-\Omega(1)}$ (see [DGJ+10]).

3.1.2 Preserving the closeness of the circuit to its approximations.

Consider some iteration of the restriction algorithm, in which we start with a circuit C_i of depth i, and replace it by a circuit C_{i-1} of depth $i-1$ that only *approximates* C_i. (In particular, C_i and C_{i-1} disagree on more inputs than the number of inputs in the final subcube of living variables in the end of the entire restriction process.) Recall that C_{i-1} was obtained by replacing very biased gates in C_i with corresponding constants.

Our goal now is to show how to choose subsequent restrictions such that with high probability C_i and C_{i-1} will remain close even after applying these restrictions. We will in fact choose each restriction ρ such that the following holds: For each gate Φ that was replaced by a constant $\sigma \in \{-1,1\}$, with probability $1 - \frac{1}{\mathrm{poly}(n)}$ over choice of restriction ρ it holds that $\Phi\restriction_\rho$ is still $\frac{1}{\mathrm{poly}(n)}$-close to σ (i.e., $\Pr_x[\Phi\restriction_\rho(x) \neq \sigma] \leq \frac{1}{\mathrm{poly}(n)}$; the claim that C_i and C_{i-1} remain close with high probability follows by a union-bound on the gates). Specifically, we prove that if an LTF Φ is, say, n^{-200}-close to a constant σ, and a restriction ρ is chosen such that the distribution of values for the fixed variables is n^{-200}-*pseudorandom for LTFs*, then with probability $1 - n^{-10}$ it holds that $\Phi\restriction_\rho$ is n^{-10}-close to σ (see Lemma 5.10). [7]

A natural approach to prove such a statement is the following. For any fixed choice of a set $I \subseteq [n]$ of variables to keep alive, we want to choose the values for the fixed

[6] This observation was communicated to us by Rocco Servedio, and is attributed to Li-Yang Tan.

[7] Since each gate is initially $\exp(-n^{\Omega(1)})$-close to a constant, we can afford a constant number of losses in the polynomial power in the "closeness" parameter throught the execution of the restriction algorithm.

variables from a *distribution that "fools" a test that checks whether or not* $\Phi|_\rho$ *is close to* σ. That is, consider a test $T : \{-1,1\}^{[n]\setminus I} \to \{-1,1\}$ that gets as input values $z \in \{-1,1\}^{[n]\setminus I}$ for the fixed variables $[n] \setminus I$, and decides whether or not Φ remains close to σ in the subcube corresponding to $\rho = \rho_{I,z}$. When z is chosen uniformly, with high probability $\Phi|_\rho$ remains close to σ, and hence the acceptance probability of T is high; thus, any distribution over $\{-1,1\}^{[n]\setminus I}$ that is pseudorandom for T also yields, with high probability, values $z \in \{-1,1\}^{[n]\setminus I}$ such that $\Phi|_{\rho_{I,z}}$ remains close to σ. The *problem with this approach* is that a test T for such a task above might be very inefficient, since it needs to evaluate Φ on all points in the subcube corresponding to $\rho = \rho_{I,z}$; thus, we might not be able to construct a pseudorandom generator with short seed to "fool" such a "complicated" test.

To solve this problem, we use the following general technique that was introduced in our previous work [Tel17], which is called *randomized tests*. Loosely speaking, a lemma from our previous work implies the following: Assume that there exists a *distribution* $\mathbf{T}$ *over tests* $\{-1,1\}^{[n]\setminus I} \to \{-1,1\}$ such that for every *fixed input* z for which $\Phi|_{\rho_{I,z}}$ is n^{-100}-close to σ it holds that $\mathbf{T}(z) = -1$, with high probability, and for every *fixed input* z for which $\Phi|_{\rho_{I,z}}$ is not n^{-10}-close to σ it holds that $\mathbf{T}(z) = 1$, with high probability. That is, the distribution $\mathbf{T}$ constitutes a "randomized test" that distinguishes, with high probability, between "excellent" z's (such that $\Phi|_{\rho_{I,z}}$ is very close to σ) and "bad" z's (such that $\Phi|_{\rho_{I,z}}$ is relatively far from σ). Also assume that almost all tests $T : \{-1,1\}^{[n]\setminus I} \to \{-1,1\}$ in the support of $\mathbf{T}$ are "fooled" by a pseudorandom generator G. Then, with high probability over choice of seed for the pseudorandom generator G, the generator outputs z such that $\Phi|_{\rho_{I,z}}$ is n^{-10}-close to σ (see Lemma 5.12 for a precise and general statement). The main point is that the distribution $\mathbf{T}$, which may have very high entropy, is *only part of the analysis*; the actual algorithm that generates $\mathbf{z}$ is simply the pseudorandom generator G.

The distribution $\mathbf{T}$ that we will use is equivalent to the following random process: Given $z \in \{-1,1\}^{[n]\setminus I}$, uniformly sample $\mathrm{poly}(n)$ points in the subcube corresponding to $\rho_{I,z}$, and accept z if Φ evaluates to the constant σ on all the sample points. We show how to construct such a distribution $\mathbf{T}$ such that *almost all* of the residual deterministic tests $T \in \mathtt{support}(\mathbf{T})$ are *conjunctions of* $p(n) = \mathrm{poly}(n)$ *LTFs*, and have very high acceptance probability (at least $1 - 1/\mathrm{poly}(p(n))$). Thus, any distribution that is $(1/\mathrm{poly}(n))$-pseudorandom for LTFs is also $(1/\mathrm{poly}(n))$-pseudorandom for almost all tests in the support of $\mathbf{T}$ (for details see the proof of Lemma 5.13). Combining this statement with the aforementioned general lemma, we deduce the following: If whenever we fix variables we choose the values for the fixed variables according to a distribution that is $(1/\mathrm{poly}(n))$-pseudorandom for LTFs, then with high probability the circuit C_i will remain close to the circuit C_{i-1}.

3.2 Reduction of standard derandomization to quantified derandomization

Given a $\mathcal{TC}^0$ circuit C of depth d over m input bits, our goal is to construct a $\mathcal{TC}^0$ circuit C' of depth $d' > d$ over $n = \mathrm{poly}(m)$ input bits such that if C accepts (resp., rejects) at least $2/3$ of its inputs then C' accepts (resp., rejects) all but $B(n) = 2^{n^{0.99}}$ of its inputs. [8] The circuit C' will use its input in order to sample inputs for C by a seeded extractor, and then compute the majority of the evaluations of C on these inputs. Specifically, fixing an extractor $E : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ for min-entropy $k = n^{0.99}$, [9] the circuit C' gets input

[8] Throughout the overview we will be somewhat informal with respect to the precise parameter values, e.g. we will use the value $B(n) = 2^{n^{0.99}}$ instead of the more precise $B(n) = 2^{n^{1-1/5d}}$.

[9] The number $B(n)$ of exceptional inputs for C' is upper-bounded by 2^k, and we want to have $B(n) = 2^{n^{0.99}}$.

$x \in \{0,1\}^n$, and outputs the majority of the values $\{C(E(x,z)) : z \in \{0,1\}^t\}$.

The main technical challenge underlying this strategy is to construct an extractor E such that the mapping of input $x \in \{0,1\}^n$ to the 2^t outputs of the extractor on all seeds (i.e., the mapping $x \mapsto \{E(x,z)\}_{z\in\{0,1\}^t}$) can be computed by a $\mathcal{TC}^0$ circuit with *as few wires as possible*. In our construction, the seed length will be $t = 1.01 \cdot \log(n)$, and thus the number of output bits will be $2^t \cdot m \approx n^{1.01}$; we will construct a $\mathcal{TC}^0$ circuit that computes the mapping of x to these $n^{1.01}$ output bits with only a *super-linear number of wires* (i.e., the number of wires is only slightly larger than the number of output bits). Indeed, a crucial point in our construction is that we will efficiently compute the outputs of the extractor on all seeds in a "batch", rather than compute the extractor separately for each seed.

3.2.1 Our starting point: A construction of C' with $n^{3.01}$ wires

As our starting point, let us construct a suitable circuit C' that has $n^{3.01}$ wires and is based on Trevisan's extractor [Tre01]. Given an input $x \in \{0,1\}^n$ and seed $z \in \{0,1\}^t$, Trevisan's extractor first computes an encoding $\bar{x}$ of x by an $(1/m^2)$-balanced error-correcting code (i.e., a code in which every non-zero codeword has relative Hamming weight $1/2 \pm m^{-2}$). [10] Fixing a suitable combinatorial design of m sets $S_1, ..., S_m$ of size $|S_i| = \log(|\bar{x}|)$ in a universe of size t, the output of $E(x,z)$ is the m bits of $\bar{x}$ in the coordinates specified by $z{\restriction}_{S_1}, ..., z{\restriction}_{S_m}$.

An initial important observation is that the circuit C' only needs to compute the encoding $\bar{x}$ of x *once*, and then each of the 2^t copies of C can take its inputs directly from the bits of $\bar{x}$ (i.e., each copy of C corresponds to a fixed seed z, and takes its inputs from locations in $\bar{x}$ that are determined by z and by the predetermined combinatorial design). This is indeed a form of "batch computation" of the extractor on all seeds.

Let us see why this construction uses $n^{3.01}$ wires. To encode x into $\bar{x}$ we can use known polynomial-time constructions of suitable *linear* codes that map n bits to $n \cdot \mathrm{poly}(m) < n^{1.01}$ bits (e.g., [NN93, ABN+92, TS17]). Since the code is linear in $x \in \{0,1\}^n$, each bit of $\bar{x} \in \{0,1\}^{n^{1.01}}$ can be computed by a $\mathcal{TC}^0$ circuit with $n^{1.01}$ wires, and thus the number of wires that we use to compute $\bar{x}$ is $n^{2.02}$. Now, recall that we want the extractor to work for min-entropy $k = n^{0.99}$; relying on Trevisan's proof and on standard constructions of combinatorial designs, the required seed length is $t < 3 \cdot \log(n)$. [11] Therefore, the number of copies of C in C' is $2^t = n^3$, and the overall number of wires in C' is $n^{2.02} + n^3 \cdot m < n^{3.01}$.

3.2.2 The actual construction of C' with $n^{1.01}$ wires

There are two parts in the construction above that led us to use a large number of wires: First, the seed length of the extractor is $t = 3 \cdot \log(n)$, which yields $2^t = n^3$ copies of C; and secondly, the number of wires required to compute the encoding $\bar{x}$ of x is super-quadratic, rather than super-linear. Let us now describe how to handle each of these two problems, and obtain a construction with only $n^{1.01}$ wires.

To reduce the seed length t of the extractor, we follow the approach of Raz, Reingold, and Vadhan [RRV02]. They showed that Trevisan's extractor works even if we replace standard combinatorial designs by a more relaxed notion that they called *weak designs* (see

[10] Trevisan's extractor only needs a $(1/2 - O(1/m), \mathrm{poly}(m))$-list-decodable code, but we will not rely on this potential relaxation.

[11] Trevisan's proof requires a design such that $|S_i \cap S_j| \leq \log(k/2m)$ (see [Tre01, Sec. 3.3]). Relying on standard constructions of combinatorial designs (see, e.g., [Tre01, Lem. 8]), a suitable design can be constructed with a universe size of $t = e^{\ln(m)/\log(2k/m)+1} \cdot \frac{\log^2(|x|)}{\log(k/2m)} \approx 1.01 \cdot e \cdot \log(n) < 3 \cdot \log(n)$.

Definition 6.1). Indeed, weak designs can be constructed with a smaller universe size t, which yields a smaller seed length for the extractor. Their construction yields $t = 2 \cdot \log(n)$, and we show a modified construction of weak designs that for our setting of parameters yields $t = 1.01 \cdot \log(n)$ (see Lemma 6.2).

The second challenge is to construct an ϵ-balanced error-correcting code that maps n bits to $n \cdot \mathrm{poly}(1/\epsilon)$ bits, and can be computed by a $\mathcal{TC}^0$ circuit of depth d with $n^{1+O(1/d)} + n \cdot \mathrm{poly}(1/\epsilon)$ wires (this is the code that we will use to compute $\bar{x}$ from x; see Corollary 6.8). To describe the code, we describe the encoding process of $x \in \{0,1\}^n$, which has two steps: First we encode x by a code with constant rate and constant relative distance, and then perform a second encoding that amplifies the distance of the code to $1/2 - \epsilon$.

Computing a code with distance $\Omega(1)$. In the first step, we encode x by a linear error-correcting code that has distance $\Omega(1)$, instead of $1/2 - \epsilon$, and also has rate $\Omega(1)$ and can be computed in $\mathcal{TC}^0$ with $n^{1.01}$ wires. This will be done using *tensor codes* that are based on any (arbitrary) initial good linear error-correcting code.

To see why tensor codes are helpful, assume that $n = r^2$, for some $r \in \mathbb{N}$, and fix a linear code $\mathtt{ECC}$ that maps r bits to $O(r)$ bits and has constant relative distance. Thinking of the input $x \in \{0,1\}^n$ as an $r \times r$ matrix, we first encode each row of the matrix x using $\mathtt{ECC}$, to obtain an $r \times O(r)$ matrix x', and then encode each column of x' using $\mathtt{ECC}$, to obtain an $O(r) \times O(r)$ matrix $\hat{x}$. By well-known properties of tensor codes, this yields a linear error-correcting code with constant rate and constant relative distance. Moreover, computing the code in $\mathcal{TC}^0$ only requires $n^{1.51}$ wires: This is because the strings that we encode with $\mathtt{ECC}$ (which are the rows of x in the first step and then the columns of x' in the second step) are each of length $r = \sqrt{n}$. Thus, each of the $O(n)$ bits in $\hat{x}$ is a linear function of $\sqrt{n}$ bits, and the latter can be computed by $\mathcal{TC}^0$ circuit with $n^{.51}$ wires.

To obtain a code with $n^{1.01}$ wires instead of $n^{1.51}$ wires we can use a tensor code of higher order. Specifically, assume that $n = r^{d_0}$, for some large constant d_0, and think of x as a tensor of dimensions $[r]^{d_0}$. The encoding process will consist of $d_0 = O(1)$ iterations, and in each iteration we encode strings of length r in the tensor by $\mathtt{ECC}$. The final codeword will be of length $(O(r))^{d_0} = O(n)$, will have constant relative distance, and can be computed by a $\mathcal{TC}^0$ circuit with only $O(n) \cdot r^{1.01} < n^{1+2/d_0}$ wires. (See Section 6.2 for further details.)

Amplifying the distance from $\Omega(1)$ to $1/2 - \epsilon$. Assume that the previous step mapped the input $x \in \{0,1\}^n$ to $\hat{x} \in \{0,1\}^{\hat{n}}$, where $\hat{n} = O(n)$. If x was a non-zero message, then $\hat{x}$ has relative Hamming weight $\Omega(1)$. Our goal now is to increase the Hamming weight of $\hat{x}$ to $1/2 - \epsilon$, using as few wires as possible. To do so we rely on the strategy of Naor and Naor [NN93], which is based on expander random walks. (This strategy was also recently used by Ta-Shma [TS17] to construct almost-optimal ϵ-balanced codes.)

Specifically, fix a graph G on $\hat{n}$ vertices with constant degree and constant spectral gap. Associate the $\hat{n}$ vertices of G with the coordinates of $\hat{x}$, and consider a random walk on G that starts at a uniformly-chosen vertex and walks $\ell = O(\log(1/\epsilon))$ steps. With probability at least ϵ, such a walk meets the set of coordinates in which $\hat{x}$ is non-zero (since this set has constant density). Thus, if we take such a random walk on the coordinates of $\hat{x}$, and output the parity of a random subset of the bits of $\hat{x}$ that we encountered, with probability at least $1/2 - \epsilon$ we will output one.

The encoding $\bar{x}$ of $\hat{x}$ is thus the following. Every coordinate in $\bar{x}$ is associated with a specific walk W of length ℓ on G and with a subset $S \subseteq [\ell]$; thus, $\bar{x}$ has $2^{\log(n)+O(\ell)} =$

$n \cdot \mathrm{poly}(1/\epsilon)$ coordinates. The bit of $\bar{x}$ at a coordinate associated with a walk W and with a subset $S \subseteq [\ell]$ is the parity of the S bits of $\hat{x}$ encountered in the walk W. Thus, each bit in $\bar{x}$ is the parity of at most $\ell = O(\log(1/\epsilon))$ bits in $\hat{x}$, so computing $\bar{x}$ from $\hat{x}$ only requires $n \cdot \mathrm{poly}(1/\epsilon) \cdot \ell^{1.01} = n \cdot \mathrm{poly}(1/\epsilon)$ wires. Recall that in our setting we need $\epsilon = 1/m^2 = n^{-\Omega(1)}$; the number of wires is thus at most $n^{1.01}$. By the preceding paragraph, if $\hat{x}$ has Hamming weight $\Omega(1)$ then $\bar{x}$ has Hamming weight at least $1/2 - \epsilon$.

4 Preliminaries

Throughout the paper, the letter n will always denote the number of inputs to a function or a circuit. We denote random variables by boldface letters, and denote by $\mathbf{u}_n$ the uniform distribution on n bits.

We are interested in Boolean functions, represented as functions $f : \{-1,1\}^n \to \{-1,1\}$. We say that a function $f : \{-1,1\}^n \to \{-1,1\}$ accepts an input $x \in \{-1,1\}^n$ if $f(x) = -1$. For two Boolean functions f and g over a domain $\mathfrak{D}$, we say that f and g are δ-close if $\Pr_{x \in \mathfrak{D}}[f(x) = g(x)] \geq 1 - \delta$.

For a vector $w = (w_1, ..., w_n) \in \mathbb{R}^n$, we denote by $\|w\|_2$ the standard ℓ_2-norm $\|w\|_2 = \sqrt{\sum_{i \in [n]} w_i^2}$. For $h < n$, we denote $w_{>h} = (w_{h+1}, ..., w_n) \in \mathbb{R}^{n-h}$ and $w_{\geq h} = (w_h, ..., w_n) \in \mathbb{R}^{n-h+1}$. For two vectors $w, x \in \mathbb{R}^n$, we denote $\langle w, x \rangle = \sum_{i \in [n]} w_i \cdot x_i$.

4.1 Two probabilistic inequalities

We will rely on two standard facts from probability theory that assert concentration and anti-concentration bounds for certain distributions. Specifically, we will need a standard version of Hoeffding's inequality, and a corollary of the Berry-Esséen theorem:

Theorem 4.1 *(Hoeffding's inequality; for a proof see, e.g., [DP09, Sec. 1.7]). Let $w \in \mathbb{R}^n$, and let $\mathbf{z}$ be a uniformly-chosen random vector in $\{-1,1\}^n$. Then, for any $t > 0$ it holds that*

$$\Pr\left[|\langle w, \mathbf{z} \rangle| \geq t \cdot \|w\|_2 \right] \leq \exp(-\Omega(t^2)) .$$

Theorem 4.2 *(a corollary of the Berry-Esséen theorem; see, e.g., [DGJ+10, Thm 2.1, Cor 2.2]). Let $w \in \mathbb{R}^n$ and $\mu > 0$ such that for every $i \in [n]$ it holds that $|w_i| \leq \mu \cdot \|w\|_2$, and let $\mathbf{z}$ be a uniformly-chosen random vector in $\{-1,1\}^n$. Then, for any $\theta \in \mathbb{R}$ and $t > 0$ it holds that:*

$$\Pr\left[\langle w, \mathbf{z} \rangle \in \theta \pm t \cdot \|w\|_2 \right] \leq 2 \cdot (t + \mu) .$$

4.2 Linear threshold functions and circuits

A linear threshold function (or LTF, in short) $\Phi : \{-1,1\}^n \to \{-1,1\}$ is a function of the form $\Phi(x) = \mathrm{sgn}(\langle x, w \rangle - \theta)$, where $w \in \mathbb{R}^n$ is a vector of real "weights", and $\theta \in \mathbb{R}$ is a real number (the "threshold"), and $\langle x, w \rangle = \sum_{i \in [n]} x_i \cdot w_i$ denotes the standard inner-product over the reals. [12] Indeed, the majority function is the special case where the weights are identical (e.g., $w_i = 1$ for all $i \in [n]$) and the threshold is zero (i.e., $\theta = 0$).

We will be interested in linear threshold circuits, which are circuits that consist only of LTF gates with unbounded fan-in and fan-out. We assume that linear threshold circuits

[12] When dealing with LTFs we can assume, without loss of generality, that $\langle w, x \rangle \neq \theta$ for every $x \in \{-1,1\}^n$ (because for every Boolean function over $\{-1,1\}^n$ that is computable by an LTF there exists an LTF that computes the function such that $\langle w, x \rangle \neq \theta$ for every $x \in \{-1,1\}^n$).

are *layered*, in the sense that for each gate Φ, all the gates feeding into Φ have the same distance from the inputs. For $n, d, m \in \mathbb{N}$, let $\mathcal{C}_{n,d,m}$ be the class of linear threshold circuits over n input bits of depth $d \geq 1$ and with at most m wires. For some fixed sizes and depths, linear threshold circuits are known to be stronger than circuits with majority gates; however, linear threshold circuits can be simulated by circuits with majority gates with a polynomial size overhead and with one additional layer (see [GHR92, GK98]). Thus, the class $\mathcal{TC}^0$ as a whole equals the class of linear threshold circuits.

The following are standard definitions (see, e.g., [Ser07, DGJ+10]), which refer to "structural" properties of LTFS and will be useful for us throughout the paper.

Definition 4.3 *(regularity). For $\epsilon > 0$, we say that a vector $w \in \mathbb{R}^n$ is* ϵ-regular *if for every $i \in [n]$ it holds that $|w_i| \leq \epsilon \cdot \|w\|_2$. An LTF $\Phi = (w, \theta)$ is ϵ-regular if w is ϵ-regular.*

Definition 4.4 *(critical index). When $w \in \mathbb{R}^n$ satisfies $|w_1| \geq |w_2| \geq ... \geq |w_n|$, the* ϵ-critical index *of w is defined as the smallest $h \in [n]$ such that $w_{>h}$ is ϵ-regular (and $h = \infty$ if no such $h \in [n]$ exists). The critical index of an LTF $\Phi = (w, \theta)$ is the critical index of w', where $w' \in \mathbb{R}^n$ is the vector that is obtained from w by permuting the coordinates in order to have $|w'_1| \geq ... \geq |w'_n|$.*

Definition 4.5 *(balanced LTF). For $t \in \mathbb{R}$, we say that an LTF $\Phi = (w, \theta)$ is t-balanced if $|\theta| \leq t \cdot \|w\|_2$; otherwise, we say that Φ is t-imbalanced.*

Representation of linear threshold circuits The algorithm in Theorem 1.1 gets as input an explicit representation of a linear threshold circuit C, where the weights and thresholds of the LTFs in C may be arbitrary real numbers. Throughout the paper we will not be specific about how exactly C is represented as an input to the algorithm, since the algorithm works in any reasonable model. In particular, the algorithm only performs *addition, subtraction, and comparison operations* on the weights and thresholds of the LTFs in C.

Explicitly suggesting one convenient model, one may assume that the weights and threshold of each LTF are integers of unbounded magnitude (since the real numbers can be truncated at some finite precision without changing the function). In this case, the circuit C has a binary representation, and the required time to perform addition, subtraction, and comparison on these integers is linear in the representation size. [13]

4.3 Pseudorandomness

We need the following two standard definitions of pseudorandom distributions and of pseudorandom generators (or PRGs, in short).

Definition 4.6 *(pseudorandom distribution). For $\epsilon > 0$ and a domain $\mathfrak{D}$, we say that a distribution $\mathbf{z}$ over $\mathfrak{D}$ is ϵ-pseudorandom for a class of functions $\mathcal{F} \subseteq \{\mathfrak{D} \to \{-1, 1\}\}$ if for every $f \in \mathcal{F}$ it holds that $\Pr_{z \sim \mathbf{z}}[f(z) = -1] \in \Pr_{z \in \mathfrak{D}}[f(z) = -1] \pm \epsilon$.*

Definition 4.7 *(pseudorandom generator). Let $\mathcal{F} = \bigcup_{n \in \mathbb{N}} \mathcal{F}_n$, where for every $n \in \mathbb{N}$ it holds that $\mathcal{F}_n$ is a set of functions $\{-1, 1\}^n \to \{-1, 1\}$, and let $\epsilon : \mathbb{N} \to [0, 1]$ and $\ell : \mathbb{N} \to \mathbb{N}$. An algorithm G is a* pseudorandom generator for $\mathcal{F}$ with error parameter ϵ and seed length ℓ *if for every $n \in \mathbb{N}$, when G is given as input 1^n and a random seed of length $\ell(n)$, the output distribution of G is ϵ-pseudorandom for $\mathcal{F}_n$.*

[13] It is well-known that every LTF over n input bits has a representation with integer weights of magnitude $2^{\tilde{O}(n)}$ (for proof see, e.g., [Hås94]), and therefore the circuit C actually has a representation of size $\text{poly}(n)$. However, we do not know of a polynomial-time algorithm to find such a representation for a given circuit C.

We will rely on the following recent construction of a pseudorandom generator for LTFs, by Gopalan, Kane, and Meka [GKM15]:

Theorem 4.8 *(a PRG for LTFs; [GKM15, Cor. 1.2]). For every $\epsilon > 0$, there exists a polynomial-time pseudorandom generator for the class of LTFs with seed length $O\left(\log(n/\epsilon) \cdot (\log\log(n/\epsilon))^2\right)$.*

A distribution $\mathbf{z}$ over $\{-1,1\}^n$ is δ-almost t-wise independent if for every $S \subseteq [n]$ of size $|S| = t$ it holds that $\mathbf{z}_S$ is δ-close to the uniform distribution over $\{-1,1\}^t$ in statistical distance. We will need the following standard tail bound for such distributions.

Fact 4.9 *(tail bound for almost t-wise independent distributions). Let $t \geq 4$ be an even number, and let $\delta : \mathbb{N} \to [0,1]$. Let $\mathbf{x}_1, ..., \mathbf{x}_n$ be variables in $\{0,1\}$ that are $\delta(n)$-almost t-wise independent, and denote $\mu = \mathbb{E}\left[\frac{1}{n} \cdot \sum_{i\in[n]} \mathbf{x}_i\right]$. Then, for any $\zeta > 0$ it holds that $\Pr\left[\left|\frac{1}{n} \cdot \sum_{i\in[n]} \mathbf{x}_i - \mu\right| \geq \zeta\right] < 8 \cdot \left(\frac{t\cdot\mu\cdot n + t^2}{\zeta^2 \cdot n^2}\right)^{t/2} + (2\cdot n)^t \cdot \delta(n)$.*

In particular, for $t = \Theta(1)$ and $\zeta = \mu/2$ and $\delta(n) = 1/p(n)$, where $p(n)$ is a sufficiently large polynomial, we have that

$$\Pr\left[\frac{1}{n} \cdot \sum_{i\in[n]} \mathbf{x}_i \in \mu \pm (\mu/2)\right] = O\left((\mu\cdot n)^{-t/2}\right) .$$

We now define the notion of a distribution that is ϵ-pseudorandomly concentrated, and show that it is essentially equivalent to the notion of being ϵ-pseudorandom for LTFs. The equivalence was communicated to us by Rocco Servedio, and is attributed to Li-Yang Tan.

Definition 4.10 *(ϵ-pseudorandomly concentrated distribution). For $n \in \mathbb{N}$ and $\epsilon > 0$, we say that a distribution $\mathbf{z}$ over $\{-1,1\}^n$ is ϵ-pseudorandomly concentrated if the following holds: For every $w \in \mathbb{R}^n$ and every $a < b \in \mathbb{R}$ it holds that $\Pr[\langle w, \mathbf{z}\rangle \in [a,b]] \in \Pr[\langle w, \mathbf{u}_n\rangle \in [a,b]] \pm \epsilon$.*

Claim 4.11 *(being pseudorandomly concentrated is equivalent to being pseudorandom for LTFs). Let $\mathbf{z}$ be a distribution over $\{-1,1\}^n$. Then,*

1. *If $\mathbf{z}$ is ϵ-pseudorandom for LTFs, then $\mathbf{z}$ is (2ϵ)-pseudorandomly concentrated.*
2. *If $\mathbf{z}$ is ϵ-pseudorandomly concentrated, then $\mathbf{z}$ is ϵ-pseudorandom for LTFs.*

Proof. Let us first prove Item (1). Fix $w \in \mathbb{R}^n$ and $I = [a,b] \subseteq \mathbb{R}$. For any fixed $z \in \{-1,1\}^n$, exactly one of three events happens: Either $\langle w, z\rangle \in I$, or $\langle w, z\rangle < a$, or $\langle w, z\rangle > b$. Since the event $\langle w, z\rangle < a$ can be tested by an LTF (i.e., by the LTF $\Phi(z) = \mathrm{sgn}(a - \langle w, z\rangle)$), this event happens with probability $\Pr_{z\in\{-1,1\}^n}[\langle w, z\rangle < a] \pm \epsilon$ under a choice of $z \sim \mathbf{z}$. Similarly, the event $\langle w, z\rangle > b$ happens with probability $\Pr_{z\in\{-1,1\}^n}[\langle w, z\rangle > b] \pm \epsilon$ under a choice of $z \sim \mathbf{z}$. Thus, the probability under a choice of $z \sim \mathbf{z}$ that $\langle w, z\rangle \in I$ is $\Pr_{z\in\{-1,1\}^n}[\langle w, z\rangle \in I] \pm 2\epsilon$.

To see that Item (2) holds, let $\Phi = (w, \theta)$ be an LTF over n input bits, and let $M = \|w\|_1 = \sum_{i\in[n]} |w_i|$. Then, for every $z \in \{-1,1\}^n$ it holds that $\Phi(z) = -1$ if and only if $z \in [-M, \theta]$. Thus, $\Pr[\Phi(\mathbf{z}) = -1] = \Pr[\mathbf{z} \in [-M,\theta]] \in \Pr[\mathbf{u}_n \in [-M,\theta]] \pm \epsilon = \Pr[\Phi(\mathbf{u}_n) = -1] \pm \epsilon$. ∎

4.4 Restrictions

A restriction for functions $\{-1,1\}^n \to \{-1,1\}$ is a subset of $\{-1,1\}^n$. We will be interested in restrictions that are subcubes, and such restrictions can be described by a string $\rho \in \{-1,1,\star\}^n$ in the natural way (i.e., the subcube consists of all strings $x \in \{-1,1\}^n$ such that for every i such $\rho_i \neq \star$ it holds that $x_i = \rho_i$). We will sometimes describe a restriction by a pair $\rho = (I,z)$, where $I = \{i \in [n] : \rho_i = \star\}$ is the set of variables that the restriction keeps alive, and $z = (\rho_i)_{i \in ([n] \setminus I)} \in \{-1,1\}^{[n] \setminus I}$ is the sequence of values that ρ assigns to the variables that are fixed.

We identify strings $r \in \{-1,1\}^{(q+1)\cdot n}$, where $n, q \in \mathbb{N}$, with restrictions $\rho = \rho_r \in \{-1,1,\star\}^n$, as follows: Each variable is assigned a block of $q+1$ bits in the string; the variable remains alive if the first q bits in the block are all 1, and otherwise takes the value of the $(q+1)^{th}$ bit. When we refer to a "block" in the string that corresponds to a restriction, we mean a block of $q+1$ bits that corresponds to some variable. When we say that a restriction is chosen from a distribution $\mathbf{r}$ over $\{-1,1\}^{(q+1)\cdot n}$, we mean that a string is chosen according to $\mathbf{r}$, and interpreted as a restriction.

In addition, we will sometimes identify a *pair* of strings $y \in \{-1,1\}^{q\cdot n}$ and $z \in \{-1,1\}^n$ with a restriction $\rho = \rho_{y,z}$. In this case, the restriction $\rho = \rho_{y,z}$ is the restriction ρ_r that is obtained by combining y and z to a string r in the natural way (i.e., appending a bit from z to each block of q bits in y). Note that the string y determines which variables ρ keeps alive, and the string z determinse the values that ρ assigns to the fixed variables.

4.5 Seeded extractors and averaging samplers

We recall the standard definitions of seeded extractors and of averaging samplers, and state the well-known equivalence between the two. In this context it will be more convenient to represent Boolean functions as functions $\{0,1\}^n \to \{0,1\}$.

Definition 4.12 *(seeded extractors). A function $f : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ is a (k,ϵ)-*extractor *if for every distribution $\mathbf{x}$ on $\{0,1\}^n$ such that $\max_{x \in \{0,1\}^n}[\Pr[\mathbf{x} = x]] \leq 2^{-k}$ it holds that the distribution $f(\mathbf{x}, \mathbf{u}_t)$ is ϵ-close to the uniform distribution on $\mathbf{u}_m$ in statistical distance.*

Definition 4.13 *(averaging samplers). A function $f : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ is an* averaging sampler with accuracy $\epsilon > 0$ and error $\delta > 0$ *if it satisfies the following. For every $T \subseteq \{0,1\}^m$, for all but a δ-fraction of the strings $x \in \{0,1\}^n$ it holds that $\Pr_{z \in \{0,1\}^t}[f(x,z) \in T] = |T|/2^m \pm \delta$.*

Proposition 4.14 *(seeded extractors are equivalent to averaging samplers). Let $f : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$. Then, the following two assertions hold:*

1. *If f is a (k,ϵ)-extractor, then f is an averaging sampler with accuracy ϵ and error $\delta = 2^{k-n}$.*
2. *If f is an averaging sampler with accuracy ϵ and error δ, then f is an $(n - \log(\epsilon/\delta), 2\epsilon)$-extractor.*

For a proof of Proposition 4.14 see, e.g., [Vad12, Cor. 6.24]. In the current paper we will only use the first item of Proposition 4.14.

5 A quantified derandomization algorithm for linear threshold circuits

Let us now state a more general version of Theorem 1.1 and prove it.

Theorem 5.1 *(Theorem 1.1, restated). Let $d \geq 1$, let $\epsilon > 0$, and let $\delta = d \cdot 30^{d-1} \cdot \epsilon$. Then, there exists a deterministic algorithm that for every $n \in \mathbb{N}$, when given as input a circuit $C \in \mathcal{C}_{n,d,n^{1+\epsilon}}$, runs in time $n^{O(\log\log(n))^2}$, and for the parameter $B(n) = \frac{1}{10} \cdot 2^{n^{1-\delta}}$ satisfies the following:*

1. *If C accepts all but at most $B(n)$ of its inputs, then the algorithm accepts C.*
2. *If C rejects all but at most $B(n)$ of its inputs, then the algorithm rejects C.*

To obtain the parameters of Theorem 1.1, for any $d \geq 1$ let $\epsilon = 2^{-10d}$. Then, the algorithm from Theorem 5.1 works when the number of exceptional inputs of C is at most $B(n) = \frac{1}{10} \cdot 2^{n^{1-\delta}} > 2^{n^{1-1/5d}}$. The deterministic algorithm from Theorem 5.1 is based on the following *pseudorandom restriction algorithm*, whose construction and proof appear in Section 5.1.

Proposition 5.2 *(pseudorandom restriction algorithm). Let $d \geq 1$, let $\epsilon > 0$ be a sufficiently small constant, and let $\delta = d \cdot 30^{d-1} \cdot \epsilon$. Then, there exists a polynomial-time algorithm that for every $n \in \mathbb{N}$, when given as input a circuit $C \in \mathcal{C}_{n,d,n^{1+\epsilon}}$ and a random seed of length $O(\log(n) \cdot (\log\log(n))^2)$, with probability at least $1 - n^{-\epsilon/2}$ satisfies the following:*

1. *The algorithm outputs a restriction $\rho \in \{-1, 1, \star\}^n$ that keeps at least $n^{1-\delta}$ variables alive.*
2. *The algorithm outputs an LTF $\Phi : \{-1,1\}^{\rho^{-1}(\star)} \to \{-1,1\}$ such that Φ is $1/10$-close to $C\restriction_\rho$ (i.e., $\Pr_{x \in \{-1,1\}^{\rho^{-1}(\star)}}[C(x) = \Phi(x)] \geq 9/10$).*

Let us now prove the main result (i.e., Theorem 5.1) relying on Proposition 5.2.

Proof of Theorem 5.1. We iterate over all seeds for the algorithm from Proposition 5.2. For each seed that yields both a restriction ρ that keeps at least $n^{1-\delta}$ variables alive and an LTF Φ over $\{-1,1\}^{\rho^{-1}(\star)}$, we estimate the acceptance probability of Φ up to an error of $\frac{1}{5}$; this is done by iterating over the seeds of the pseudorandom generator from Theorem 4.8 (instantiated with error parameter $1/5$). If for most of the seeds, our estimate of the acceptance probability of Φ is at least $\frac{3}{5}$, then we accept C; and otherwise we reject C. The running time of the algorithm is $2^{O(\log(n) \cdot (\log\log(n))^2)} = n^{O(\log\log(n))^2}$.

Recall that all but $O(n^{-\epsilon})$ of the seeds yield ρ and Φ such that ρ keep at least $n^{1-\delta} > \log(10 \cdot B(n))$ variables alive and such that Φ is $1/10$-close to $C\restriction_\rho$; we call such seeds *good* seeds. Now, if C accepts all but at most $B(n)$ inputs, then for every good seed, the acceptance probability of $C\restriction_\rho$ is at least $9/10$, and thus the acceptance probability of Φ is at least $\frac{4}{5}$, which implies that our estimate of the latter will be at least $3/5$. Thus, the algorithm will accept C. On the other hand, if C rejects all but at most $B(n)$ inputs, then by a similar argument for all good seeds it holds that the estimate of the acceptance probability of Φ will be at most $2/5$, and thus the algorithm will reject C. ■

5.1 Pseudorandom restriction algorithm

We prove Proposition 5.2 in three steps. The first step, in Section 5.1.1, will be to prove that a suitably-chosen pseudorandom restriction turns any single LTF to be very biased, with high probability. The second step, in Section 5.1.2, will leverage the first step to construct an algorithm that gets as input a linear threshold circuit, and applies pseudorandom restrictions to reduce the depth of the circuit by one layer. And the final step, in Section 5.1.3, will be to iterate the construction of the second step in order to prove Proposition 5.2.

5.1.1 Pseudorandom restrictions and a single LTF

As mentioned in the introduction, an illustrative example for the effects of restrictions on LTFs is the majority function $\Phi(x) = \mathrm{sgn}(\sum_{i\in[n]} x_i)$. For $p \in (0,1)$, denote by $\boldsymbol{\mathcal{R}}_p$ the distribution of restrictions on n variables such that for every $i \in [n]$ independently it holds that the i^{th} variable remains alive with probability p, and is otherwise assigned a uniform random bit. Then, we have the following:

Fact 5.3 *(a random restriction and the majority function). Let $\Phi(x) = \mathrm{sgn}(\sum_{i\in[n]} x_i)$, and let $p = n^{-\Omega(1)}$. Then, for every $t \geq 1$, with probability at least $1 - O(t \cdot \sqrt{p})$ over $\rho \sim \boldsymbol{\mathcal{R}}_p$ it holds that $\Phi\restriction_\rho$ is t-imbalanced*

Proof. Let $I \subseteq [n]$ be the set of variables that ρ keeps alive. With probability $1 - \exp(-n^{\Omega(1)})$ it holds that $\|w_I\|_2 \in \sqrt{pn} \pm \sqrt{pn}/2$. Conditioned on $\|w_I\|_2 \leq 2 \cdot \sqrt{pn}$, it also holds that $\left\|w_{[n]\setminus I}\right\|_2 \geq \sqrt{n}/2$, which implies that for every $i \in ([n] \setminus I)$ it holds that $|w_i| = 1 \leq (2/\sqrt{n}) \cdot \left\|w_{[n]\setminus I}\right\|_2$. In this case, by the Berry-Esséen theorem (i.e., by Theorem 4.2), for any $t \geq 1$, the probability that $\left\langle w_{[n]\setminus I}, z_{[n]\setminus I}\right\rangle$ falls in the interval $\pm 4t \cdot \sqrt{p} \cdot \left\|w_{[n]\setminus I}\right\|_2$ (which contains the interval $\pm t \cdot \|w_I\|_2$) is at most $O(t \cdot \sqrt{p} + \frac{2}{\sqrt{n}}) = O(t \cdot \sqrt{p})$. ■

Our goal in this section is to prove a statement that is similar to Fact 5.3, but that holds for an *arbitrary LTF* Φ, and holds also when the restriction ρ is sampled pseudorandomly, rather than uniformly. For simplicity, we only state the proposition informally at the moment (for a formal statement see Proposition 5.8):

Proposition 5.4 *(pseudorandom restriction lemma for a single LTF; informal). Let $n \in \mathbb{N}$, let $p = n^{-\Omega(1)}$, and let $t = p^{-\Omega(1)}$. Let $\mathbf{y}$ be a distribution over $\{-1,1\}^{\log(1/p)\cdot n}$ that is p-almost $O(\log(1/p))$-wise independent, and let $\mathbf{z}$ be a distribution over $\{-1,1\}^n$ that is $p^{\Omega(1)}$-pseudorandomly concentrated. Then, for any LTF Φ over n input bits, the probability over choice of restriction $\rho \sim (\mathbf{y}, \mathbf{z})$ that $\Phi\restriction_\rho$ is t-balanced is at most $p^{\Omega(1)}$.*

A high-level description of the proof. Let $\Phi = (w, \theta)$ be an LTF over n input bits, and without loss of generality assume that $|w_1| \geq |w_2| \geq ... \geq |w_n|$. Denote by $I \subseteq [n]$ the set of variables that ρ keeps alive, and by $z_{[n]\setminus I} \in \{-1,1\}^{[n]\setminus I}$ the values that ρ assigns to the fixed variables. Then, the restricted function is of the form $\Phi\restriction_\rho = \left(w_I, \theta - \left\langle w_{[n]\setminus I}, z_{[n]\setminus I}\right\rangle\right)$, and the restricted function is t-balanced if and only if the sum $\left\langle w_{[n]\setminus I}, z_{[n]\setminus I}\right\rangle$ falls in the interval $\theta \pm 2t \cdot \|w_I\|_2$. Our goal will be to show that this event is unlikely.

The proof is based on a modification of the case analysis that appears in [CSS16, Lem. 34, Sec. 4.2, Apdx. C.]. Specifically, for the parameter values $\mu = \Omega(1/t)$ and $k = \tilde{O}(t^2)$, we will consider two separate cases.

Case 1: The μ-critical index of Φ is at most k. Let $h \leq k$ be the μ-critical index of Φ, and denote $T = [n] \setminus [h]$. We first claim that with probability $1 - p^{\Omega(1)}$ over choice of $y \sim \mathbf{y}$ it holds that $\|w_I\|_2 \leq p^{\Omega(1)} \cdot \|w_T\|_2$. This is the case since with probability at least $1 - h \cdot p = 1 - p^{\Omega(1)}$, all the first h variables are fixed by ρ, and since the expected value of $\|w_{I\cap T}\|_2$ is $\sqrt{p} \cdot \|w_T\|_2$.

Condition on any fixed choice of $y \sim \mathbf{y}$ such that $\|w_I\|_2 \leq p^{\Omega(1)} \cdot \|w_T\|_2$. We will prove that with probability $1 - p^{\Omega(1)}$ over a *uniform choice of* $z \in \{-1,1\}^n$ it holds that

$\left\langle w_{[n]\setminus I}, z_{[n]\setminus I} \right\rangle$ does not fall in the interval $\theta \pm t \cdot p^{\Omega(1)} \cdot \|w_T\|_2$ (which contains the interval $\theta \pm t \cdot \|w_I\|_2$, due to our fixed choice of y). Since $\mathbf{z}$ is $p^{\Omega(1)}$-pseudorandomly concentrated, it will follow that this event also holds with probability $1 - p^{\Omega(1)}$ under a choice of $z \sim \mathbf{z}$.

To prove the claim about a uniform choice of $z \in \{-1,1\}^n$, condition *any arbitrary fixed values* $z_{[h]} \in \{-1,1\}^h$ for the first h variables. Then, the probability that $\left\langle w_{[n]\setminus I}, z_{[n]\setminus I} \right\rangle$ falls in the interval $\theta \pm t \cdot p^{\Omega(1)} \cdot \|w_T\|_2$ (which is what we want to bound) equals the probability that $\langle w_{T\setminus I}, z_{T\setminus I} \rangle_2$ falls in the interval $\theta' \pm t \cdot p^{\Omega(1)} \cdot \|w_T\|_2$, where $\theta' = \theta - \left\langle w_{[h]}, z_{[h]} \right\rangle$. Since h is the μ-critical index of w we have that w_T is μ-regular; also, since $\|w_I\|_2 \leq p^{\Omega(1)} \cdot \|w_T\|_2$ (due to our choice of y), it follows that $w_{T\setminus I}$ is also (2μ)-regular and that $\|w_T\|_2 \approx \|w_{T\setminus I}\|_2$. By the Berry-Esséen theorem, the probability that $\langle w_{T\setminus I}, z_{T\setminus I} \rangle$ falls in an interval of length $t \cdot p^{\Omega(1)} \cdot \|w_{T\setminus I}\|_2$ is at most $O(t \cdot p^{\Omega(1)} + \mu) = p^{\Omega(1)}$ (see Lemma 5.5).

Case 2: The μ-critical index of Φ is larger than k. Similarly to the previous case, with probability at least $1 - p^{\Omega(1)}$ it holds that all the first k variables are fixed by ρ. Condition on any fixed $y \sim \mathbf{y}$ that fixes all the first k variables. What we will show is that with high probability over $z \sim \mathbf{z}$, the sum $\left\langle w_{[n]\setminus I}, \mathbf{z}_{[n]\setminus I} \right\rangle$ falls outside the interval $\theta \pm (1/4\mu) \|w_{>k}\|_2$, which contains the interval $\theta \pm t \cdot \|w_I\|_2$ (since $I \subseteq ([n] \setminus [k])$ and $\mu = \Omega(1/t)$).

As before, we first analyze the case in which z is chosen uniformly in $\{-1,1\}^n$. To do so we rely on a lemma of Servedio [Ser07], which asserts that the weights in w decrease exponentially up to the critical index. Intuitively, since the critical index is large (i.e., more than k), the exponential decay of the weights implies that $\|w_{>k}\|_2$ is small. Thus, when uniformly choosing $z \in \{-1,1\}^n$, the sum $\left\langle w_{[n]\setminus I}, z_{[n]\setminus I} \right\rangle$ is unlikely to fall in the small interval $\theta \pm (1/4\mu) \cdot \|w_{>k}\|_2$; specifically, this happens with probability at most $\mu = p^{\Omega(1)}$ (see Claim 5.7.1 for a precise statement).

Since the event $\left\langle w_{[n]\setminus I}, z_{[n]\setminus I} \right\rangle \in \theta \pm (1/4\mu) \cdot \|w_{>k}\|_2$ happens with probability $p^{\Omega(1)}$ when $z \in \{-1,1\}^n$ is chosen uniformly, and the distribution $\mathbf{z}$ is $p^{\Omega(1)}$-pseudorandomly concentrated, the event also happens with probability at most $p^{\Omega(1)}$ over a choice of $z \sim \mathbf{z}$.

The full proof. We will first prove an auxiliary lemma, which analyzes the effect of uniformly-chosen restrictions on regular LTFs (see Lemma 5.5). Then, we will prove a version of Proposition 5.4 that only holds for LTFs with *bounded critical index* (see Lemma 5.6), and a version of Proposition 5.4 that only holds for LTFs with *large critical index* (see Lemma 5.7). Finally, we will formally state a more general version of Proposition 5.4 and prove it (see Proposition 5.8).

The following auxiliary lemma considers a regular vector $w \in \mathbb{R}^m$, a fixed set of variables $I \subseteq [m]$ that will be kept alive, and a uniformly-chosen assignment $z \in \{-1,1\}^m$ for the fixed variables. The lemma will be used in the proof of Lemma 5.6.

Lemma 5.5 *(pseudorandom restriction lemma for regular LTFs). Let $m \in \mathbb{N}$, let $\mu \in (0,1)$, and let $\lambda \leq 3/4$. Let $w' \in \mathbb{R}^m$ be a μ-regular vector, and let $I \subseteq [m]$ such that $\|w'_I\|_2 < \lambda \cdot \|w'\|_2$. Then, for any $\theta' \in \mathbb{R}$ and $t > 0$, the probability over uniform choice of $z \in \{-1,1\}^m$ that $\left\langle w'_{[m]\setminus I}, z_{[m]\setminus I} \right\rangle \in \theta' \pm t \cdot \lambda \cdot \|w'\|_2$ is at most $O(t \cdot \lambda + \mu)$.*

Proof. Note that $\left\|w'_{[m]\setminus I}\right\|_2^2 > \|w'\|_2^2/4$; this is the case because $\|w'_I\|_2^2 < \lambda \cdot \|w'\|_2^2 \leq \frac{3}{4} \cdot$

$\|w'\|_2^2$. It follows that $w'_{[m]\setminus I}$ is 2μ-regular, since for every $i \in [m]$ we have that $\left|w'_i\right| \leq \mu \cdot \|w'\|_2 \leq 2\mu \cdot \left\|w'_{[m]\setminus I}\right\|_2$. It also follows that the interval $\theta \pm t \cdot \lambda \cdot \|w'\|_2$ is contained in the interval $\theta \pm 2t \cdot \lambda \cdot \left\|w'_{[m]\setminus I}\right\|_2$. By the Berry-Esséen theorem (i.e., by Theorem 4.2), the probability over a uniform choice of $z \in \{-1,1\}^m$ that the sum $\left\langle w_{[m]\setminus I}, z_{[m]\setminus I}\right\rangle$ falls in a fixed interval of length $2t \cdot \lambda \cdot \left\|w_{[m]\setminus I}\right\|$ is at most $O(t \cdot \lambda + \mu)$. ■

The following lemma asserts that a suitably-chosen pseudorandom restriction turns every LTF with *bounded critical index* to be very biased, with high probability. The specific parameters that are chosen for the lemma will be useful for us when proving the general case (i.e., Proposition 5.8, which holds for arbitrary LTFs).

Lemma 5.6 *(pseudorandom restriction lemma for LTFs with small critical index). Let $n \in \mathbb{N}$, let $p \in [0,1]$ be a power of two, let $c \in \mathbb{N}$ be a constant, and let $t \leq p^{-1/(3c-2)}$ and $\mu = 1/4t^c$. Let $\mathbf{y}$ be a distribution over $\{-1,1\}^{\log(1/p)\cdot n}$ that is p-almost $O(\log(1/p))$-wise independent, and let $\mathbf{z}$ be a distribution over $\{-1,1\}^n$ that is μ-pseudorandomly concentrated. Then, for any LTF Φ over n input bits with μ-critical index at most $k = 10^3 \cdot \mu^{-2} \cdot \log^2(1/\mu)$, the probability over choice of $\rho \sim (\mathbf{y}, \mathbf{z})$ that $\Phi\restriction_\rho$ is t-balanced is at most $\tilde{O}(t^{1+c/2}) \cdot \sqrt{p} + O(t^{-c})$.*

Proof. Let $\Phi = (w, \theta)$ be an LTF gate over n input bits with critical index $h \leq k$, and without loss of generality assume that $|w_1| \geq |w_2| \geq ... \geq |w_n|$. Let $I \subseteq [n]$ be the random variable that is the set of live variables under $\mathbf{y}$; then, it holds that:

Claim 5.6.1. *With probability at least $1 - O(\mu + p \cdot k)$ over $y \sim \mathbf{y}$ it holds that $I \subseteq ([n] \setminus [h])$ and that $\|w_I\|_2 \leq \sqrt{p/\mu} \cdot \left\|w_{[n]\setminus[h]}\right\|_2$.*

Proof. Since $\mathbf{y}$ is p-almost $O(\log(1/p))$-wise independent, each variable is kept alive with probability at most $2p$. Thus, the probability over $y \sim \mathbf{y}$ that the first h variables are all fixed is at least $1 - 2p \cdot h$. Also, the expected value of $\left\|w_{I\cap([n]\setminus[h])}\right\|_2^2$ is at most $2p \cdot \left\|w_{[n]\setminus[h]}\right\|_2^2$, and hence with probability at least $1 - 2\mu$ it holds that $\left\|w_{I\cap([n]\setminus[h])}\right\|_2 \leq \sqrt{p/\mu} \cdot \left\|w_{[n]\setminus[h]}\right\|_2$. By a union-bound, with probability at least $1 - O(\mu + p \cdot h) > 1 - O(\mu + p \cdot k)$ it holds that $I \subseteq ([n] \setminus [h])$ and that $\|w_I\|_2 = \left\|w_{I\cap([n]\setminus[h])}\right\|_2 \leq \sqrt{p/\mu} \cdot \left\|w_{[n]\setminus[h]}\right\|_2$. □

Fix any $y \sim \mathbf{y}$ such that the first h variables are all fixed, and such that $\|w_I\|_2 \leq \sqrt{p/\mu} \cdot \left\|w_{[n]\setminus[h]}\right\|_2$. Our goal will be to prove that with high probability over $z \sim \mathbf{z}$ it holds that $\left\langle w_{[n]\setminus I}, \mathbf{z}_{[n]\setminus I}\right\rangle \notin \theta \pm t \cdot \sqrt{p/\mu} \cdot \left\|w_{[n]\setminus[h]}\right\|_2$; this suffices to prove the lemma, since $t \cdot \sqrt{p/\mu} \cdot \left\|w_{[n]\setminus[h]}\right\| \geq t \cdot \|w_I\|_2$. To do so, we first analyze the setting in which $z \in \{-1,1\}^n$ is chosen uniformly, rather than from the distribution $\mathbf{z}$:

Claim 5.6.2. *The probability over a uniform choice of $z \in \{-1,1\}^n$ that $\left\langle w_{[n]\setminus I}, z_{[n]\setminus I}\right\rangle \in \theta \pm t \cdot \sqrt{p/\mu} \cdot \left\|w_{[n]\setminus[h]}\right\|_2$ is at most $O(t \cdot \sqrt{p/\mu} + \mu)$.*

Proof. The claim is trivial for $\mu \leq 2p$, so it suffices to prove the claim under the assumption that $\mu > 2p$. Condition on any arbitrary assignment $z_{[h]} \in \{-1,1\}^h$ for the first h variables, and note that the vector $w_{>h} \in \{-1,1\}^{n-h}$ is μ-regular (since h is the μ-critical index of Φ).

Let $T = [n] \setminus [h]$. Observe that when conditioning on $z_{[h]}$, the event $\left\langle w_{[n]\setminus I}, z_{[n]\setminus I} \right\rangle \in \theta \pm t \cdot \sqrt{p/\mu} \cdot \left\| w_{[n]\setminus[h]} \right\|_2$ happens if and only if the event $\langle w_{T\setminus I}, z_{T\setminus I} \rangle \in \theta' \pm t \cdot \sqrt{p/\mu} \cdot \|w_T\|_2$ happens, where $\theta' = \theta - \left\langle w_{[h]}, z_{[h]} \right\rangle$. Since w_T is μ-regular, we can invoke Lemma 5.5 with $w' = w_T$ and with $\lambda = \sqrt{p/\mu} \leq 3/4$ (the inequality is since $\mu > 2p$), and deduce the probability of the event $\langle w_{T\setminus I}, z_{T\setminus I} \rangle \in \theta' \pm t \cdot \sqrt{p/\mu} \cdot \|w_T\|_2$ is at most $O(t \cdot \sqrt{p/\mu} + \mu)$. □

Since $\mathbf{z}$ is μ-pseudorandomly concentrated, it follows from Claim 5.6.2 that the probability over $z \sim \mathbf{z}$ that $\left\langle w_{[n]\setminus I}, \mathbf{z}_{[n]\setminus I} \right\rangle \in \theta \pm t \cdot \sqrt{p/\mu} \cdot \left\| w_{[n]\setminus[h]} \right\|_2$ is at most $O(t \cdot \sqrt{p/\mu} + \mu)$. Thus, the probability over choice of $\rho \sim (\mathbf{y}, \mathbf{z})$ that $\Phi\restriction_\rho$ is t-balanced is at most $O(t \cdot \sqrt{p/\mu} + \mu + p \cdot k) = \tilde{O}(t^{1+c/2}) \cdot \sqrt{p} + O(t^{-c})$, where the last equality relied on the hypothesis that $t \leq p^{-1/(3c-2)}$. ■

The following lemma is similar to Lemma 5.6, but holds for LTFs with *large critical index*.

Lemma 5.7 *(pseudorandom restriction lemma for LTFs with large critical index). Let $n \in \mathbb{N}$, let $p \in [0,1]$ be a power of two, and let $\mu > 0$. Let $\mathbf{y}$ be a distribution over $\{-1,1\}^{\log(1/p)\cdot n}$ that is p-almost $O(\log(1/p))$-wise independent, and let $\mathbf{z}$ be a distribution over $\{-1,1\}^n$ that is μ-pseudorandomly concentrated. Then, for any LTF Φ over n input bits with μ-critical index larger than $k = 10^3 \cdot \mu^{-2} \cdot \log^2(1/\mu)$, the probability over choice of $\rho \sim (\mathbf{y}, \mathbf{z})$ that $\Phi\restriction_\rho$ is $(1/4\mu)$-balanced is $\tilde{O}(\mu^{-2}) \cdot p + O(\mu)$.*

Proof. Let $\Phi = (w, \theta)$ be an LTF gate over n input bits with μ-critical index larger than k, and without loss of generality assume that $|w_1| \geq |w_2| \geq ... \geq |w_n|$. Also, let $I \subseteq [n]$ be the random variable that is the set of live variables under $\mathbf{y}$. Note that the probability over $y \sim \mathbf{y}$ that $I \cap [k] \neq \emptyset$ is at most $2p \cdot k = \tilde{O}(\mu^{-2}) \cdot p$ (since $\mathbf{y}$ keeps each variable alive with probability at most $2p$).

Condition on any arbitrary $y \sim \mathbf{y}$ such that $[k] \cap I = \emptyset$. Our goal now is to show that the probability over $z \sim \mathbf{z}$ that $\Phi\restriction_\rho$ is $(1/4\mu)$-balanced is $O(\mu)$. We will actually prove a stronger claim: We will show that with probability at least $1 - O(\mu)$ it holds that $\left\langle w_{[n]\setminus I}, \mathbf{z}_{[n]\setminus I} \right\rangle \notin \theta \pm (1/4\mu) \cdot \|w_{>k}\|_2$ (this claim is stronger, since $I \subseteq ([n] \setminus [k])$, which implies that $\|w_{>k}\|_2 \geq \|w_I\|_2$). To prove this assertion we will rely on the following claim, which is essentially from [CSS16, Prop. 45] and generalizes [DGJ+10, Lem. 5.8]. (Since the proof is sketched in [CSS16], we include a full proof.)

Claim 5.7.1. *Let $\mu > 0$, let $r \in \mathbb{N}$, and let $k_{r,\mu} = \frac{4r \cdot \ln(3/\mu^2)}{\mu^2}$. Let $\Phi = (w, \theta)$ be an LTF over n input bits with μ-critical index larger than $k_{r,\mu}$ such that $|w_1| \geq ... \geq |w_n|$, and let $J \subseteq [n]$ such that $J \supseteq [k_{r,\mu}]$. Then, the probability under uniform choice of $z \in \{0,1\}^n$ that $\langle w_J, z_J \rangle \in \theta \pm (1/4\mu) \cdot \left\| w_{>k_{r,\mu}} \right\|_2$ is at most 2^{-r}.*

Proof. Since the critical index of Φ is larger than $k_{r,\mu}$, a lemma of Servedio [Ser07, Lem. 3] asserts that for any $1 \leq i < j \leq k_{r,\mu}$ it holds that

$$|w_j| \leq \left\| w_{\geq j} \right\|_2 \leq \left(1 - \mu^2\right)^{(j-i)/2} \cdot \|w_{\geq i}\|_2 \leq \left(1 - \mu^2\right)^{(j-i)/2} \cdot |w_i|/\mu \,. \tag{5.1}$$

(For an equivalent statement of the lemma see [DGJ+10, Lem. 5.5].) In particular, fixing $\gamma = \frac{2\ln(3/\mu^2)}{\mu^2}$, for any $i \in \mathbb{N}$ such that $i \cdot \gamma < k_{r,\mu}$ it holds that $|w_{i\cdot\gamma}| < |w_1|/3^i$.

Let $R = 1, \gamma, ..., r \cdot \gamma < k_{r,\mu}$, and consider any arbitrary fixed value of $z_{J \setminus R}$. Then, by a claim of Diakonikolas *et al.* [DGJ+10, Clm. 5.7], there exists at most a single value $z_R \in \{-1,1\}^r$ such that $\langle w_R, z_R \rangle \in \left(\theta - \left\langle w_{J \setminus R}, z_{J \setminus R} \right\rangle\right) \pm |w_{r \cdot \gamma}|/4$. Thus, the probability under a uniform choice of $z \in \{0,1\}^n$ that $\langle w_J, z_J \rangle \in \theta \pm |w_{r \cdot \gamma}|/4$ is at most 2^{-r}.

The claim follows since $\left\| w_{>k_{r,\mu}} \right\|_2 \leq \left\| w_{\geq (r+1) \cdot \gamma} \right\|_2 \leq \mu \cdot |w_{r \cdot \gamma}|$, where the first inequality is since $k_{r,\mu} > (r+1) \cdot \gamma$ and the second inequality is due to Eq. (5.1). □

We invoke Claim 5.7.1 with the value $r = \log(1/\mu)$ and with the set $J = [n] \setminus I$, while noting that the critical index of Φ is indeed larger than $k \geq k_{r,\mu}$. Since the interval $\theta \pm (1/4\mu) \cdot \|w_{>k}\|_2$ is contained in the interval $\theta \pm (1/4\mu) \cdot \left\| w_{>k_{r,\mu}} \right\|_2$ (because $k \geq k_{r,\mu}$), we deduce that the event $\left\langle w_{[n] \setminus I}, z_{[n] \setminus I} \right\rangle \in \theta \pm (1/4\mu) \cdot \|w_{>k}\|_2$ happens with probability at most μ under a uniform choice of $z \in \{0,1\}^n$. Since $\mathbf{z}$ is μ-pseudorandomly concentrated, this event happens with probability at most $O(\mu)$ also under a choice of $z \sim \mathbf{z}$. ■

Finally, we are ready to state a more general version of Proposition 5.4 and to prove it. The proof will rely on Lemmas 5.6 and 5.7.

Proposition 5.8 *(pseudorandom restriction lemma for an arbitrary LTF). Let $n \in \mathbb{N}$, let $p \in [0,1]$ be a power of two, let $c \in \mathbb{N}$ be a constant, and let $t \leq p^{-1/(3c-2)}$. Let $\mathbf{y}$ be a distribution over $\{-1,1\}^{\log(1/p) \cdot n}$ that is p-almost $O(\log(1/p))$-wise independent, and let $\mathbf{z}$ be a distribution over $\{-1,1\}^n$ that is $(1/4t^c)$-pseudorandomly concentrated. Then, for any LTF Φ over n input bits, the probability over choice of $\rho \sim (\mathbf{y}, \mathbf{z})$ that $\Phi\restriction_\rho$ is t-balanced is at most $\tilde{O}(t^{1+c/2}) \cdot \sqrt{p} + O(t^{-c})$.*

To obtain the parameters that were stated in Section 3.1, invoke Proposition 5.8 with $c = 2$. (When $c = 2$, the hypothesis that $t \leq p^{-1/(3c-2)} = p^{-1/4}$ is not required, since for $t > p^{-1/4}$ the probability bound in the lemma's statement is trivial.)

Proof of Proposition 5.8. Let $\Phi = (w, \theta)$ be an LTF gate over n input bits, let $\mu = 1/4t^c$, and let $k = 10^3 \cdot \mu^{-2} \cdot \log^2(1/\mu)$. If the μ-critical index of Φ is at most k, the asserted probability bound follows immediately from Lemma 5.6. On the other hand, if the μ-critical index of Φ is larger than k, we can rely on Lemma 5.7. The lemma asserts that the probability that $\Phi\restriction_\rho$ is $(1/4\mu)$-balanced is at most $\tilde{O}(\mu^{-2}) \cdot p + O(\mu) < \tilde{O}(t^{1+c/2}) \cdot \sqrt{p} + O(t^{-c})$, where the inequality relies on the hypothesis that $t \leq p^{-1/(3c-2)}$. Since $(1/4\mu) \geq t$, whenever $\Phi\restriction_\rho$ is $(1/4\mu)$-imbalanced it is also t-imbalanced. ■

5.1.2 Pseudorandom restriction algorithm for a "layer" of LTFs

The next step is to construct a pseudorandom restriction algorithm that transforms a depth-d linear threshold circuit into a depth-$(d-1)$ linear threshold circuit. The key part in this step is an application of Proposition 5.8.

Proposition 5.9 *(pseudorandom restriction algorithm for a "layer" of LTFs). For every three constants $d \geq 2$ and $\epsilon > 0$ and $c > 0$, there exists a polynomial-time algorithm that gets as input a circuit $C \in \mathcal{C}_{n,d,n^{1+\epsilon}}$ and a random seed of length $O(\log(n) \cdot (\log\log(n))^2)$, and with probability at least $1 - n^{-\epsilon}$ outputs the following:*

1. *A restriction $\rho \in \{-1,1,\star\}^n$ that keeps at least $n' = \Omega(n^{1-24 \cdot \epsilon})$ variables alive.*

2. *A circuit* $\widetilde{C} \in \mathcal{C}_{n',d-1,(n')^{1+30\epsilon}}$ *that agrees with* C *on at least* $1 - n^{-c}$ *of the inputs in the subcube that corresponds to* ρ *(i.e.,* $\Pr_{x \in \{-1,1\}^{|\rho^{-1}(\star)|}}[C\restriction_\rho(x) = \widetilde{C}(x)] > 1 - n^{-c}$*).*

High-level overview of the proof. The key step of the algorithm is to apply Proposition 5.8 with parameters $p = n^{-\beta}$ and $c = 1$ and $t = p^{-1/5}$, where $\beta = O(\epsilon)$. The lemma asserts that, in expectation, all but approximately $n^{-\beta/5}$ of the gates will become t-imbalanced (for simplicity, ignore polylogarithmic factors for now). Such imbalanced gates are extremely close to a constant function, so we can replace the gates by the corresponding constants and get a circuit that agrees with the original circuit on almost all inputs.

As for the other $n^{-\beta/5}$-fraction of the gates, we expect that the number of wires feeding into them will decrease by a factor of p after the restriction. Specifically, assume that indeed the fan-in of each gate decreased by a factor of at least p; then, the expected number of wires feeding into the balanced gates after the restriction is at most

$$\sum_{\Phi \text{ gate}} \Pr[\Phi \text{ balanced}] \cdot p \cdot (\#\text{ wires incoming to } \Phi) \leq n^{-\beta/5} \cdot p \cdot n^{1+\epsilon} \,. \tag{5.2}$$

Thus, with probability at least $1 - n^{-\beta/10}$, the number of wires feeding into balanced gates is at most $(n^{\epsilon-\beta/10}) \cdot p \cdot n$, which is much smaller than the expected number of living variables (i.e., than $p \cdot n$) if $\beta > 10\epsilon$. When this happens, we can afford to simply fix all the variables that feed into balanced gates, making those gates constant too.

The argument above relied on the assumption that the fan-in of each gate Φ decreased by a factor of at least p. We can argue that this indeed holds with high probability for all gates with fan-in at least n^α, where $\alpha > \beta$, but we will need to separately handle gates with fan-in at most n^α. This will be done in two steps: The first is an initial preprocessing step (before applying Proposition 5.8), in which we fix every variable with fan-out more than $2 \cdot n^\epsilon$; since there are at most $n^{1+\epsilon}$ wires, this step fixes at most $n/2$ variables. Then, after applying Proposition 5.8 and fixing the variables that feed into balanced gates with fan-in at least n^α, we show that there exists a set I of variables of size approximately $n^{-(\alpha+\epsilon)} \cdot (p \cdot n)$ such that after fixing all variables outside I, each gate with fan-in at most n^α has fan-in at most one (see Claim 5.10.1). Thus, we can fix the variables outside I, and then replace each gate with fan-in at most n^α with the corresponding variable (or with its negation). At this point all the gates in the bottom layer have been replaced by constants or by variables.

Proof of Proposition 5.9. Let $G = \{\Phi_1, ..., \Phi_r\}$ be the set of gates in the bottom layer of C. For $\alpha = 12\epsilon$, let $S \subseteq G$ be the set of gates with fan-in at most n^α, and let $L = G \setminus S$ be the set of gates with fan-in more than n^α.

The restriction ρ will be composed of four restrictions $\rho_1, ..., \rho_4$. When describing the construction of each restriction, we will always assume that all previous restrictions were successful (we will describe exactly what "successful" means for each restriction). Also, after each restriction, we fix additional variables if necessary, in order to obtain an exact number of living variables in the end of the step.

Let $\mathbf{z}$ be a distribution over $\{-1,1\}^n$ that is $(1/q(n))$-pseudorandom for LTFs, where q is a sufficiently large polynomial. We mention in advance that for each $i \in [4]$, the values for variables that are fixed by ρ_i will always be decided by sampling from $\mathbf{z}$.

The first restriction ρ_1: Reduce the fan-out of input gates. We sample $z \sim \mathbf{z}$, and fix all variables with fan-out more than $2 \cdot n^\epsilon$ to values according to z. Since the number of wires

between the bottom-layer gates and the input variables is at most $n^{1+\epsilon}$, and each fixing of a variable eliminates $2 \cdot n^{\epsilon}$ wires, we will fix no more than $n/2$ variables in this step. Let $n_1 = n/2$ be the number of living variables after the first step.

The second restriction ρ_2: Applying Proposition 5.8. We use Proposition 5.8 with the values $p = n^{-\beta}$, where $\beta = 11\epsilon$, and $c = 1$, and $t = p^{-1/5}$. [14] The distributions that we use are a $(1/\text{poly}(n))$-almost $O(\log(1/p))$-wise independent distribution $\mathbf{y}$ over $\{-1,1\}^{\log(1/p)\cdot n}$ and the aforementioned distribution $\mathbf{z}$ over $\{-1,1\}^n$.

Let $\mathcal{E}$ be the event in which ρ_2 keeps at least $(p \cdot n_1)/2$ variables alive, and for every gate $\Phi \in L$ it holds that $\text{fan-in}(\Phi\restriction_{\rho_2}) \leq 2p \cdot \text{fan-in}(\Phi)$. We claim that $\mathcal{E}$ happens with probability at least $1 - 1/\text{poly}(n)$. To see that this is the case, note that the expected number of living variables is $p \cdot n_1 = n^{\Omega(1)}$, and that for each gate $\Phi \in G$, the expected fan-in of $\Phi\restriction_{\rho_2}$ is $n^{\alpha-\beta} = n^{\Omega(1)}$. Since the choice of variables to keep alive is $\frac{1}{\text{poly}(n)}$-almost $O(1)$-independent, we can use Fact 4.9 to deduce that $\Pr[\mathcal{E}] \geq 1 - \frac{1}{\text{poly}(n)}$.

Now, assume without loss of generality that $L = \{\Phi_1, ..., \Phi_{r'}\}$, for some $r' \leq r$. For any $i \in [r']$, denote by $\mathcal{B}_i$ the event that Φ_i is t-balanced. Note that when conditioning on $\mathcal{E}$, the probability of each $\mathcal{B}_i$ is at most $\tilde{O}(n^{-\beta/5})$. Therefore, conditioned on $\mathcal{E}$, the expected number of wires feeding into t-balanced gates in L after the restriction is

$$\begin{aligned}
\mathbb{E}\left[\sum_{i\in[r']} \mathbf{1}_{\mathcal{B}_i} \cdot \text{fan-in}(\Phi_i\restriction_{\rho_2}) \middle| \mathcal{E}\right] &= \sum_{i\in[r']} \Pr[\mathcal{B}_i|\mathcal{E}] \cdot \mathbb{E}[\text{fan-in}(\Phi_i\restriction_{\rho_2})|\mathcal{E}, \mathcal{B}_i] \\
&\leq \sum_{i\in[r']} \tilde{O}(n^{-\beta/5}) \cdot (2p \cdot \text{fan-in}(\Phi_i)) \\
&= \tilde{O}(n^{-\beta/5}) \cdot p \cdot n^{1+\epsilon} \,.
\end{aligned}$$

Hence, conditioned on $\mathcal{E}$, the probability that the number of wires feeding into t-balanced gates in L after the restriction is more than $\tilde{O}(n^{-\beta/10}) \cdot p \cdot n^{1+\epsilon} = \tilde{O}(n^{\epsilon-\beta/10}) \cdot n^{1-\beta}$ is at most $O(n^{-\beta/10})$. We consider the restriction ρ_2 successful if $\mathcal{E}$ happens and if the number of wires between t-balanced gates in L and input gates is at most $\tilde{O}(n^{\epsilon-\beta/10}) \cdot n^{1-\beta}$. In this case, the number of currently-living variables is $n_2 = p \cdot n_1/2 = \frac{1}{4} \cdot n^{1-\beta}$.

After applying ρ_2, we replace any t-imbalanced gate $\Phi_i \in L$ with its most probable value $\sigma_i \in \{-1,1\}$. Note that by Theorem 4.1, each t-imbalanced gate Φ_i is $(\exp(-n^{\Omega(1)}))$-close to σ_i in the subcube that corresponds to the currently-living variables.

The third restriction ρ_3: Eliminate L-gates that remained unbiased. In this step we sample $z \sim \mathbf{z}$ again, and fix all the variables that feed into t-balanced gates according to z. Assuming that ρ_2 was successful, the number of such variables is at most $\tilde{O}(n^{\epsilon-\beta/10}) \cdot n^{1-\beta} = o(n_2)$, where we used the fact that $\beta > 10\epsilon$. Denote the restriction applied in this step by ρ_3, and note that the number of living variables after applying ρ_3 is $n_3 = \Omega(n_2) = \Omega(n^{1-11\epsilon})$.

Our goal now is to claim that for each gate Φ_i that was replaced by a constant $\sigma \in \{-1,1\}$ prior to applying ρ_3, it still holds that Φ_i is close to σ in the subcube $\{-1,1\}^{\rho_3^{-1}(\star)}$. To do so we will rely on a lemma that asserts the following: If an LTF Φ_i is δ-close to a

[14] For simplicity, we assume that $p = n^{-11\epsilon}$ is a power of two. Otherwise, we can choose β to be a value very close to 11ϵ such that p will be a power of two, with no meaningful change to the rest of the proof (the proof only relies on the fact that $10\epsilon < \beta < \alpha$).

constant function, then with probability $1-\gamma$ over choice of $z \sim \mathbf{z}$ it holds that $\Phi_i|_\rho$ is δ'-close to the same constant function, as long as $\delta \leq \text{poly}(\delta', \gamma)$ and that $\mathbf{z}$ is $\text{poly}(\gamma)$-pseudorandom for LTFs.

Lemma 5.10 *(bias preservation lemma). Let $n \in \mathbb{N}$, and let $\delta, \delta', \gamma > 0$ such that $\delta \leq (\gamma \cdot \delta')^{10}$. Let $\Phi = (w, \theta)$ be an LTF over n input bits that is δ-close to a constant function $\sigma \in \{-1, 1\}$, let $I \subseteq [n]$, and let $\mathbf{z}$ be a distribution over $\{-1,1\}^{[n]\setminus I}$ that is $(\delta' \cdot \gamma^2)$-pseudorandom for LTFs. Then, with probability $1 - O(\gamma)$ over choice of $z \sim \mathbf{z}$ it holds that $\Phi|_{(I,z)}$ is δ'-close to σ.*

The proof of Lemma 5.10 is deferred to Section 5.2. We invoke Lemma 5.10 with I being the set of variables that are kept alive by ρ_3, and $\delta = \exp(-n^{\Omega(1)})$, and $\gamma = 1/\text{poly}(n)$, and $\delta' = n^{-10\cdot(2+4\epsilon+c)}$. After union-bounding over at most $r \leq n^{1+\epsilon}$ gates that were replaced by constants, with probability $1 - 1/\text{poly}(n)$ it holds that all these gates are δ'-close to constants in the subcube $\{-1,1\}^{\rho_3^{-1}(\star)}$.

The fourth restriction ρ_4: Eliminate gates with small fan-in. We will rely on the following claim, which is an algorithmic version of [CSS16, Prop. 36]:

Claim 5.10.1. *For $k' = 2 \cdot n^{\alpha+\epsilon}$, we can deterministically find in $\text{poly}(n)$ time a set I of at least n_3/k' living variables such that when fixing all variables not in I to any arbitrary values, the fan-in of each gate in S is at most one.*

Proof. Consider the graph in which the vertices are the input gates $x_1, ..., x_{n_3}$, and two vertices x_i and x_j are connected (in the graph) if and only if there exists a gate $\Phi_i \in S$ that is connected (in the circuit) to both x_i and x_j. Note that this graph has degree at most k', since every living variable has fan-out at most $2 \cdot n^\epsilon$, and every gate in S has fan-in at most n^α. Therefore, we can greedily construct an independent set I in the graph of size at least n_3/k', which is indeed the set of variables that we wanted. □

The algorithm finds a set I using Claim 5.10.1, samples $z \sim \mathbf{z}$, and fixes all the variables outside I according to z. This yields a restriction that reduces the fan-in of each gate in S to one. Thus, each gate $\Phi \in S$ now simply takes the value of an input gate (or its negation), which implies that the gates that are connected to Φ (in the layer above it) can be connected immediately to the corresponding input gate, and we can remove Φ from the circuit. The number of living variables is $n_4 = n_3/k' = \Omega(n^{1-24\epsilon})$.

To conclude, we claim that the gates that were previously replaced by constants are still close to constants in the new subcube. This is done by invoking Lemma 5.10 with I being the aforementioned set of size n_4, and with parameter values $\delta = n^{-10\cdot(2+4\epsilon+c)}$, and $\gamma = n^{-(1+3\epsilon)}$, and $\delta' = n^{-(c+1+\epsilon)}$. After union-bounding over the gates that were replaced by constants, with probability at least $1 - n^{-2\epsilon}$ it holds that all these gates are δ'-close to constants in the final subcube. It follows that the original circuit is δ''-close to the new circuit in the final subcube, where $\delta'' \leq \delta' \cdot n^{1+\epsilon} \leq n^{-c}$.

Accounting for the parameters. We obtained a circuit in $\tilde{C} \in \mathcal{C}_{n_4, d-1, n^{1+\epsilon}}$. Since $n^{1+\epsilon} = O(n_4^{\frac{1+\epsilon}{1-24\epsilon}}) < n_4^{(1+\epsilon)(1+25\epsilon)} \leq n_4^{1+30\epsilon}$, we have that $\tilde{C} \in \mathcal{C}_{n_4, d-1, n_4^{1+30\epsilon}}$. To sample the restriction $\rho = \rho_4 \circ ... \circ \rho_1$, we sampled from the distribution $\mathbf{z}$ four times, and from the distribution $\mathbf{y}$ a single time. A sample from $\mathbf{y}$ can obtained with seed length $O(\log(n))$, and relying on Theorem 4.8, each sample from $\mathbf{z}$ can be obtained with seed length $O(\log(n) \cdot (\log\log(n))^2)$.

Finally, let us account for the error probability. The first step is deterministic and always succeeds. In the second step, the algorithm is unable to simplify the circuit if the event $\mathcal{E}$ does not happen, or if the number of wires between t-balanced gates in L and input gates is too large. Denoting the latter event by $\mathcal{E}'$, the probability of error is at most $\Pr[\neg\mathcal{E}] + \Pr[\mathcal{E}'|\mathcal{E}] \leq O(n^{-\beta/10})$. The last type of error to account for is the probability that $\tilde{C}$ is not n^{-c}-close to C in $\{-1,1\}^{\rho^{-1}(\star)}$; as detailed above, this happens with probability at most $n^{-2\epsilon}$. The overall error is thus $O(n^{-\beta/10} + n^{-2\epsilon}) < n^{-\epsilon}$. ■

5.1.3 Pseudorandom restriction algorithm for linear threshold circuits

We are now ready to construct the pseudorandom restriction algorithm that simplifies any linear threshold circuit to a single LTF gate (i.e., Proposition 5.2). The proof will consist of $d-1$ applications of Proposition 5.9. In each application, we will use Lemma 5.10 to claim that all the approximations in previous applications of Proposition 5.9 still hold.

Proposition 5.11 *(Proposition 5.2, restated). Let $d \geq 1$, let $\epsilon > 0$ be a sufficiently small constant, and let $\delta = d \cdot 30^{d-1} \cdot \epsilon$. Then, there exists a polynomial-time algorithm that for every $n \in \mathbb{N}$, when given as input a circuit $C \in \mathcal{C}_{n,d,n^{1+\epsilon}}$ and a random seed of length $O(\log(n) \cdot (\log\log(n))^2)$, with probability at least $1 - n^{-\epsilon/2}$ satisfies the following:*

1. *The algorithm outputs a restriction $\rho \in \{-1,1,\star\}^n$ that keeps at least $n^{1-\delta}$ variables alive.*
2. *The algorithm outputs an LTF $\Phi : \{-1,1\}^{\rho^{-1}(\star)} \to \{-1,1\}$ such that Φ is $1/10$-close to $C\restriction_\rho$ (i.e., $\Pr_{x\in\{-1,1\}^{\rho^{-1}(\star)}}[C(x) = \Phi(x)] \geq 9/10$).*

Proof. We repeatedly invoke Proposition 5.9, for $d-1$ times. For $i \in [d-1]$, let $\rho^{(i)}$ be the restriction that is obtained in the i^{th} invocation of Proposition 5.9, and let $\rho = \rho^{(d-1)} \circ ... \circ \rho^{(1)}$ be the final restriction. Let $C_0 = C$, and for $i \in [d-1]$, let C_i be the circuit that is obtained after the i^{th} invocation of Proposition 5.9. Also let $\epsilon_0 = \epsilon$ and $\epsilon_i = 30 \cdot \epsilon_{i-1} = 30^i \cdot \epsilon$, and let $n_0 = n$ and $n_i = \Omega\left((n_{i-1})^{1-24\epsilon_{i-1}}\right)$.

We say that an invocation of Proposition 5.9 is *successful* if the two items in the proposition's statement are satisfied (i.e., the algorithm outputs a restriction that keeps sufficiently many live variables, and a circuit of smaller depth that agrees with the original circuit on almost all inputs). Assuming all invocations of Proposition 5.9 are successful, for each $i \in [d-1]$ it holds that $C_i \in \mathcal{C}_{n_i,d-i,n_i^{1+\epsilon_i}}$, and in particular C_{d-1} is a single LTF Φ. Also, in this case, the number of living variables after all invocations is

$$n_{d-1} = n^{\Pi_{i=0}^{d-2}(1-24\epsilon_i)} > n^{1-24\cdot\sum_{i=0}^{d-2}\epsilon_i} > n^{1-24\cdot d\cdot\epsilon_{d-2}} > n^{1-\delta} . \tag{5.3}$$

The required seed length for the $d-1$ invocations of Proposition 5.9 is $\tilde{O}(\log(n))$. To bound the probability of error, for each $i \in [d-1]$, assume that all previous $i-1$ invocations were successful, and note that the probability that the i^{th} invocation of Proposition 5.9 fails is at most $n_{i-1}^{-\epsilon_{i-1}} < (n^{1-\delta})^{-\epsilon}$ (the inequality is since we assumed that the previous invocations of Proposition 5.9 were successful, which implies that $n_{i-1} \geq n^{1-\delta}$, by a calculation similar to Eq. (5.3)). Thus, the accumulated probability of error is at most $d \cdot (n^{1-\delta})^{-\epsilon} < n^{-\epsilon/2}$, where the inequality relied on the fact that ϵ is sufficiently small.

Condition on all the $d-1$ invocations of Proposition 5.9 being successful. Recall that in this case, for every $i \in [d-1]$ it holds that C_i is n^{-c}-close to $C_{i-1}\restriction_{\rho^{(i)}}$; we now claim that, with high probability, this approximation continues to hold even in the subcube that corresponds to the final restriction ρ.

Claim 5.11.1. *For every $i \in [d-1]$, with probability $1 - 1/\mathrm{poly}(n)$ it holds that $(C_{i-1})\restriction_\rho$ is $1/10d$-close to $(C_i)\restriction_\rho$.*

Proof. For each $j \in \{i, ..., d-1\}$, recall that $\rho^{(j)}$ is the composition of four restrictions, denoted by $\rho_1^{(j)}, ..., \rho_4^{(j)}$. Fix $i \in [d-1]$, condition on any fixed choice for $\rho_1^{(i)}$ and $\rho_2^{(i)}$, and let $C' = (C_{i-1})\restriction_{\rho_1^{(i)}, \rho_2^{(i)}}$. Recall that immediately after applying $\rho_2^{(i)}$, the algorithm from Proposition 5.9 replaces a set of $m \leq n^{1+\epsilon_{d-(i-1)}}$ LTF gates, denoted $\Phi_1, ..., \Phi_m$, with a corresponding set of constants $\sigma_1, ..., \sigma_m \in \{-1, 1\}$. Let $\widetilde{C}'$ be the circuit that is obtained from C' by the aforementioned replacement. Finally, note that for every choice of final restriction ρ it holds that $(C_{i-1})\restriction_\rho = C'\restriction_\rho$ and $(C_i)\restriction_\rho = \widetilde{C}'\restriction_\rho$.

Our goal now will be to show that for every fixed $k \in [m]$, with probability $1 - 1/\mathrm{poly}(n)$ over choice of ρ it holds that $(\Phi_k)\restriction_\rho$ is $1/(10dm)$-close to σ_k. This suffices to conclude the proof, since it follows (by a union-bound over the m gates) that with probability $1 - 1/\mathrm{poly}(n)$, for every $k \in [m]$ it holds that $(\Phi_k)\restriction_\rho$ is $1/(10dm)$-close to σ_k; and whenever the latter event happens we have that $C'\restriction_\rho$ is $1/(10d)$-close to $\widetilde{C}'\restriction_\rho$.

Towards the aforementioned goal, fix $k \in [m]$, and recall that Φ_k is δ_0-close to some constant function $\sigma_k \in \{-1, 1\}$, where $\delta_0 = \exp\left(n_{i-1}^{-\Omega(1)}\right) = \exp\left(n^{-\Omega(1)}\right)$, where the inequality is since $n_{i-1} = n^{\Omega(1)}$ (recall that we conditioned on all invocations of Proposition 5.9 being successful). Observe that the final restriction ρ is composed of $t \stackrel{\text{def}}{=\!=} 4 \cdot (d - i - 1) + 2$ additional restrictions on the domain of Φ_k: Two additional restrictions $\rho_3^{(i)}$ and $\rho_4^{(i)}$ in the i^{th} invocation of Proposition 5.9, and for each $j \in \{i+1, ..., d-1\}$, four restrictions $\rho_1^{(j)}, ..., \rho_4^{(j)}$ in the j^{th} invocation of Proposition 5.9. Recall that each of the t restrictions is chosen by first choosing (deterministically or pseudorandomly) a set of variables to keep alive, and then *independently* choosing values for the fixed variables. Therefore, we will now repeatedly use Lemma 5.10, to claim that each restriction preserves the closeness of Φ_k to σ_k.

For convenience, rename the t restrictions $\rho_3^{(i)}, \rho_4^{(i)}, \rho_1^{(i+1)}, ..., \rho_4^{(i+1)}, ..., \rho_1^{(d-1)}, ..., \rho_4^{(d-1)}$, and denote them by $\rho'^{(1)}, ..., \rho'^{(t)}$. Let $\gamma = n^{-c}$ for a sufficiently large constant $c > 1$. Note that $\delta_0 < n^{-10^{2t} \cdot c}$, and for every $r \in [t]$ let $\delta_r = \delta_{r-1}^{1/10^2}$; it follows that for every $r \in [t]$ it holds that $\delta_{r-1} \leq (\gamma \cdot \delta_r)^{10}$. We prove by induction on $r \in [t]$ that with probability at least $1 - O(n^{-c})$ it holds that $(\Phi_k)\restriction_{\rho'^{(1)} \circ ... \circ \rho'^{(r)}}$ is δ_r-close to σ_k. For the base case $r = 1$ we rely on the hypothesis that Φ_k is δ_0-close to σ_k, and use Lemma 5.10 with the values $\delta = \delta_0$ and $\delta' = \delta_1$ and $\gamma = n^{-c}$ as above. For the induction step $r > 1$, we condition on $(\Phi_k)\restriction_{\rho'^{(1)} \circ ... \circ \rho'^{(r-1)}}$ being δ_{r-1}-close to σ_k, and again use Lemma 5.10 with the values $\delta = \delta_{r-1}$ and $\delta' = \delta_r$ and $\gamma = n^{-c}$. Hence, with probability at least $1 - O(n^{-c})$ it holds that $(\Phi_k)\restriction_\rho$ is δ_t-close to σ_k, where $\delta_t = n^{-c} < 1/(10dm)$. □

Thus, with probability $1 - 1/\mathrm{poly}(n)$, for every $i \in [d-1]$ it holds that $(C_{i-1})\restriction_\rho$ is $1/10d$-close to $(C_i)\restriction_\rho$. Whenever this holds, by a union-bound it follows that $C\restriction_\rho = (C_0)\restriction_\rho$ is $1/10$-close to $(C_{d-1})\restriction_\rho = C_{d-1} = \Phi$. ■

5.2 Proof of the bias preservation lemma

In this section we prove Lemma 5.10. Loosely speaking, the lemma asserts that an LTF Φ that is close to a constant $\sigma \in \{-1, 1\}$ remains close to σ when the domain is restricted by a restriction ρ in which the values for the fixed variables are chosen from a distribution that

is pseudorandom for LTFs. For the proof we will need the following lemma from [Tel17, Lem. 15] (the original notations are adapted for the current context).

Lemma 5.12 *(randomized tests). Let $n \in \mathbb{N}$, and let $\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4, \epsilon_5 > 0$ be error parameters.*

- *Let $G \subseteq \{-1,1\}^n$, and let $E \subseteq G$ such that $\Pr_{z \in \{-1,1\}^n}[z \in E] \geq 1 - \epsilon_1$.*
- *Let $\mathbf{T}$ be a distribution over functions $T : \{-1,1\}^n \to \{-1,1\}$ such that for every $z \in E$ it holds that $\Pr_{T \sim \mathbf{T}}[T(z) = -1] \geq 1 - \epsilon_2$, and for every $z \notin G$ it holds that $\Pr_{T \sim \mathbf{T}}[T(z) = 1] \geq 1 - \epsilon_3$.*
- *Let $\mathbf{z}$ be a distribution that is ϵ_5-pseudorandom for all but an ϵ_4-fraction of the tests in $\mathbf{T}$; that is, the probability over $T \sim \mathbf{T}$ that $\left|\Pr[T(\mathbf{u}_n) = -1] - \Pr[T(\mathbf{z}) = -1]\right| > \epsilon_5$ is at most ϵ_4.*

Then, the probability that $\mathbf{z} \in G$ is at least $1 - (\epsilon_1 + \epsilon_2 + \epsilon_3 + 2\epsilon_4 + \epsilon_5)$.

Fix a set $I \subseteq [n]$ of variables that the restriction keeps alive. Relying on Lemma 5.12, the proof idea for Lemma 5.10 is to design a distribution $\mathbf{T}$ over tests that gets as input $z \in \{-1,1\}^{[n] \setminus I}$, and tests whether or not Φ is close to σ in the subcube corresponding to the restriction $\rho = \rho_{I,z}$.

Lemma 5.13 *(Lemma 5.10, restated). Let $n \in \mathbb{N}$, and let $\delta, \delta', \gamma > 0$ such that $\delta \leq (\gamma \cdot \delta')^{10}$. Let $\Phi = (w, \theta)$ be an LTF over n input bits that is δ-close to a constant function $\sigma \in \{-1,1\}$, let $I \subseteq [n]$, and let $\mathbf{z}$ be a distribution over $\{-1,1\}^{[n] \setminus I}$ that is $(\delta' \cdot \gamma^2)$-pseudorandom for LTFs. Then, with probability $1 - O(\gamma)$ over choice of $z \sim \mathbf{z}$ it holds that $\Phi\restriction_{(I,z)}$ is δ'-close to σ.*

A high-level description of the proof. For every $z \in \{-1,1\}^{[n] \setminus I}$, consider the corresponding subcube $\mathfrak{C}_z = \{y \in \{-1,1\}^n : \forall i \in ([n] \setminus I), y_i = z_i\}$. Our goal is to show that with high probability over $z \sim \mathbf{z}$ it holds that Φ is close to σ in $\mathfrak{C}_z$. To do so, we will construct a distribution $\mathbf{T}$ of tests such that for any fixed $z \in \{-1,1\}^{[n] \setminus I}$, the distribution $\mathbf{T}(z)$ is equivalent to the following random process: Sample $t = \text{poly}(n)$ random points $y^{(1)}, ..., y^{(t)}$ in $\mathfrak{C}_z$, and accept if and only if $\Phi(y^{(i)}) = \sigma$ for every $i \in [t]$.

To construct the distribution $\mathbf{T}$, for every $x \in \{-1,1\}^{|I|}$ we define a corresponding test T_x as follows: The test T_x gets input $z \in \{-1,1\}^{[n] \setminus I}$, extends z to an n-bit string $y \in \{-1,1\}^n$ using the values specified in x (i.e., $y_i = x_i$ for every $i \in I$, and $y_i = z_i$ otherwise), and accepts z if and only if $\Phi(y) = \sigma$. Observe that T_x simply computes an LTF of its input z (see Eq. (5.4)). Also note that for any *fixed input* $z \in \{-1,1\}^{[n] \setminus I}$, a uniform choice of $x \in \{-1,1\}^{|I|}$ yields a uniform point $y \in \mathfrak{C}_z$. Each test in $\mathbf{T}$ corresponds to a tuple $\bar{x} = (x^{(1)}, ..., x^{(t)}) \in \{-1,1\}^{t \cdot |I|}$, and computes the function $T_{\bar{x}}(z) = \wedge_{i \in [t]} T_{x^{(i)}}(z)$.

Assume that Φ is initially δ-close to σ, for $\delta \leq 1/\text{poly}(n)$. We say that an input $z \in \{-1,1\}^{[n] \setminus I}$ is excellent if Φ is $\sqrt{\delta}$-close to σ in $\mathfrak{C}_z$, and we say that z is bad if Φ is not δ'-close to σ in $\mathfrak{C}_z$, where $\delta' = \delta^{\Omega(1)}$. Let E be the set of excellent inputs, and let B be the set of bad inputs. If we choose the parameter t (i.e., the number of sample points) such that $\frac{O(\log(n))}{\delta'} < t < \frac{1}{\sqrt{\delta} \cdot \text{poly}(n)}$, then the distribution $\mathbf{T}$ accepts every $z \in E$ with probability $1 - 1/\text{poly}(n)$, and rejects every $z \in B$, with probability $1 - 1/\text{poly}(n)$.

What remains to show is that a distribution $\mathbf{z}$ that is $(1/\text{poly}(n))$-pseudorandom for LTFs is also $(1/\text{poly}(n))$-pseudorandom for almost all tests in the support of $\mathbf{T}$. To do so, note that almost all inputs $z \in \{-1,1\}^{[n] \setminus I}$ are excellent, and each excellent input is accepted with high probability by a random test $T_{\bar{x}} \sim \mathbf{T}$. Thus, almost all of the residual

deterministic tests $T_{\bar{x}}$ in the support of $\mathbf{T}$ *accept almost all of their inputs*; in particular, at least $1 - 1/\text{poly}(n)$ of the residual tests have acceptance probability at least $1 - 1/\text{poly}(n)$. Every such test is the conjunction of $t = \text{poly}(n)$ LTFs, and each of these LTFs has acceptance probability at least $1 - 1/\text{poly}(n)$. By a union-bound over the t LTFs, the acceptance probability of such $T_{\bar{x}}$ under $\mathbf{z}$ is also $1 - t \cdot (1/\text{poly}(n)) = 1 - 1/\text{poly}(n)$.

Proof of Lemma 5.13. Without loss of generality, assume that Φ is δ-close to the constant $\sigma = -1$. For any Boolean function f over a domain $\mathfrak{D}$, let $\texttt{acc}(f) = \Pr_{x \sim \mathfrak{D}}[f(x) = -1]$. Also, denote $J = [n] \setminus I$ and $n' = |J|$, and for any $z \in \{0,1\}^{n'}$, denote by ρ_z the restriction $\rho_z = (I, z)$ (i.e., we suppress I in the notation ρ_z, since I is fixed).

Let $G = \left\{ z \in \{0,1\}^{n'} : \texttt{acc}(\Phi\restriction_{\rho_z}) \geq 1 - \delta' \right\}$. Our goal is to show that $\Pr_{z \sim \mathbf{z}}[z \in G] \geq 1 - O(\gamma)$. Let $E = \left\{ z \in \{0,1\}^{n'} : \texttt{acc}(\Phi\restriction_{\rho_z}) \geq 1 - \sqrt{\delta} \right\}$. Note that when $z \in \{-1,1\}^{n'}$ is chosen uniformly it holds that $\mathbb{E}_{z \in \{-1,1\}^{n'}} \left[\texttt{acc}(\Phi\restriction_{\rho_z}) \right] = \Pr_{x \in \{-1,1\}^n}[\Phi(x) = -1] \geq 1 - \delta$. Therefore, $\Pr_{z \in \{-1,1\}^{n'}}[z \in E] \geq 1 - \sqrt{\delta}$.

We now construct a distribution $\mathbf{T}$ over tests $\{-1,1\}^{n'} \to \{-1,1\}$ that distinguishes, with high probability, between $z \in E$ and $z \notin G$. For $x \in \{0,1\}^{|I|}$, let T_x be the function that gets as input $z \in \{0,1\}^{n'}$, and outputs the value $\Phi(y)$, where $y_J = z$ and $y_I = x$. Note that for any fixed $z \in \{-1,1\}^{n'}$, when uniformly choosing $x \in \{-1,1\}^{|I|}$ it holds that $\Pr[T_x(z) = -1] = \texttt{acc}(\Phi\restriction_{\rho_z})$. Also, T_x is an LTF of its input z, because

$$T_x(z) = \text{sgn}\left(\langle y, w \rangle - \theta\right) = \text{sgn}\left(\langle z, w_J \rangle - (\theta - \langle x, w_I \rangle)\right) \ . \tag{5.4}$$

For $t = O\left(\frac{\log(1/\gamma)}{\delta'}\right)$ and $\bar{x} = (x^{(1)}, ..., x^{(t)}) \in \{0,1\}^{t \cdot |I|}$, let $T_{\bar{x}} : \{-1,1\}^{n'} \to \{-1,1\}$ be the function such that $T_{\bar{x}}(z) = -1$ if and only if for every $i \in [t]$ it holds that $T_{x^{(i)}}(z) = -1$ (i.e., $T_{\bar{x}}$ is the conjunction $\wedge_{i \in [t]} T_{x^{(i)}}$). Our distribution $\mathbf{T}$ is the uniform distribution over the set $\left\{ T_{\bar{x}} : \bar{x} \in \{0,1\}^{t \cdot |I|} \right\}$. Observe that:

- For any fixed $z \in E$ it holds that $\Pr_{T_{\bar{x}} \sim \mathbf{T}}\left[T_{\bar{x}}(z) = -1\right] \geq 1 - t \cdot \sqrt{\delta}$.
- For any fixed $z \notin G$ it holds that $\Pr_{T_{\bar{x}} \sim \mathbf{T}}\left[T_{\bar{x}}(z) = -1\right] \leq \gamma$.

We want to show that almost all of the tests $\{T_{\bar{x}}\}_{\bar{x} \in \{0,1\}^{t \cdot |I|}}$ in the support of $\mathbf{T}$ accept almost all of their inputs. To see that this is the case, observe that

$$\mathbb{E}_{\bar{x}}\left[\texttt{acc}(T_{\bar{x}})\right] = \Pr_{\bar{x}, z}[T_{\bar{x}}(z) = -1] \geq \Pr_{z}[z \in E] \cdot \min_{z \in E}\left\{ \Pr_{\bar{x}}[T_{\bar{x}}(z) = -1] \right\} \ ,$$

which is lower-bounded by $1 - \xi^2$, where $\xi^2 = (t+1) \cdot \sqrt{\delta}$. Therefore, the fraction of tests $T_{\bar{x}}$ that reject more than ξ of their inputs is at most ξ.

Now, let $T_{\bar{x}}$ be a test such that $\texttt{acc}(T_{\bar{x}}) \geq 1 - \xi$. Since $T_{\bar{x}}$ is a conjunction of $T_{x^{(1)}}, ..., T_{x^{(t)}}$, for each $i \in [t]$ it holds that $\texttt{acc}(T_{x^{(i)}}) \geq 1 - \xi$. Also, for each $i \in [t]$ it holds that $\mathbf{z}$ is η-pseudorandom for $T_{x^{(i)}}$, where $\eta \leq (\gamma^2 \cdot \delta')$, and therefore $\Pr_{z \sim \mathbf{z}}[T_{x^{(i)}}(z) = -1] \geq 1 - \xi - \eta$. It follows that $\Pr_{z \sim \mathbf{z}}[T_{\bar{x}}(z) = -1] \geq 1 - t \cdot (\xi + \eta)$.

We invoke Lemma 5.12 with the parameters $\epsilon_1 = \sqrt{\delta}$, $\epsilon_2 = t \cdot \sqrt{\delta}$, $\epsilon_3 = \gamma$, $\epsilon_4 = \xi$, and

$\epsilon_5 = t \cdot (\xi + \eta)$, and deduce that

$$\begin{aligned}\Pr_{z\sim\mathbf{z}}[z \notin G] &\leq (t+1)\cdot\sqrt{\delta} + \gamma + 2\cdot\sqrt{t+1}\cdot\delta^{1/4} + t\cdot(\sqrt{t+1}\cdot\delta^{1/4} + \eta) \\ &= O\left(\gamma + t^{3/2}\cdot\delta^{1/4} + t\cdot\eta\right) \\ &= O\left(\gamma + (\gamma\cdot\delta')^{-3/2}\cdot\delta^{1/4} + \eta/(\gamma\cdot\delta')\right)\ ,\end{aligned}$$

which is $O(\gamma)$ since $\eta \leq (\gamma^2\cdot\delta')$ and by our hypotheses regarding γ, δ, and δ'. ■

6 Reduction of standard derandomization to quantified derandomization

In this section we prove Theorem 1.2. The core of the proof is the construction of a suitable averaging sampler (equivalently, seeded extractor) that is computable by a $\mathcal{TC}^0$ circuit with a super-linear number of wires. We therefore start by describing this construction. In the current section, as in Section 4.5, it will be more convenient to represent Boolean functions as functions $\{0,1\}^n \to \{0,1\}$, rather than $\{-1,1\}^n \to \{-1,1\}$.

In Section 6.1 we recall the definition of weak combinatorial designs, and construct such designs that are suitable for our parameter setting. In Section 6.2 we show how to compute a code with distance $1/2 - o(1)$ by a $\mathcal{TC}^0$ circuit with a super-linear number of wires. In Section 6.3 we combine the two preceding ingredients to construct an averaging sampler in $\mathcal{TC}^0$. Finally, in Section 6.4 we prove Theorem 1.2.

6.1 Weak combinatorial designs for Trevisan's extractor

Let us recall the notion of weak combinatorial designs, which was introduced by Raz, Reingold, and Vadhan [RRV02].

Definition 6.1 *(weak designs). For positive integers $m, \ell, t \in \mathbb{N}$ and an integer $\rho > 1$, an (m, ℓ, t, ρ)* weak design *is a collection of sets $S_1, ..., S_m \subseteq [t]$ such that for every $i \in [m]$ it holds that $|S_i| = \ell$ and $\sum_{j<i} 2^{|S_i \cap S_j|} \leq (m-1)\cdot\rho$.*

Raz, Reingold, and Vadhan [RRV02] showed a construction of weak designs with universe size $t = \left\lceil \frac{\ell}{\ln(\rho)} \right\rceil \cdot \ell$. In our parameter setting we will have $\log(\rho) \approx 0.99 \cdot \ell$, and for such value the construction in [RRV02] yields $t = 2 \cdot \ell$. We want to have $t \approx 1.01 \cdot \ell$, and therefore now show a more refined construction.

Lemma 6.2 *(constructing weak designs). There exists an algorithm that gets as input $m \in \mathbb{N}$ and $\ell \in \mathbb{N}$ and $\rho \in \mathbb{N}$ such that $\log(\rho) = (1-\alpha)\cdot\ell$, where $\alpha \in (0, 1/4)$, and satisfies the following. The algorithm runs in time $\mathrm{poly}(m, 2^\ell)$ and outputs an (m, ℓ, t, ρ) weak design, where $t = \lceil (1+4\alpha)\cdot\ell \rceil$.*

Proof. Let $t = \lceil (1+4\alpha)\cdot\ell \rceil$. The algorithm constructs the sets $S_1, ..., S_m \subseteq [t]$ in iterations. In each iteration $i \in [m]$ the algorithm finds S_i such that $\sum_{j<i} 2^{|S_i \cap S_j|} \leq (i-1)\cdot\rho$. To do so, the algorithm initially fixes a partition of $[t]$ into ℓ blocks. The first $t - \ell$ blocks, denoted $B_1, ..., B_{t-\ell}$, are each comprised of two elements (i.e., for $j \in [t-\ell]$ it holds that $B_j = \{2j-1, 2j\}$). The remaining $2\ell - t$ blocks, denoted $B_{t-\ell+1}, ..., B_\ell$, each consist of a single element (i.e., for $j \in \{t-\ell+1, ..., \ell\}$ it holds that $B_j = \{t-\ell+j\}$).

For $i \in [m]$, let us describe the i^{th} iteration, after $S_1, ..., S_{i-1}$ were already chosen in previous iterations. Consider a set S_i that is chosen by independently choosing one random element from each of the ℓ blocks to include in S_i. [15] For $j \in [i-1]$ and $k \in [\ell]$, let $Y_{j,k}$ be the indicator variable of whether the element from the k^{th} block that is included in S_j is also included in S_i (i.e., $Y_{j,k} = 1$ iff $B_k \cap S_j \cap S_i \neq \emptyset$). Note that for $k \neq k' \in [m]$ it holds that $Y_{j,k}$ and $Y_{j,k'}$ are independent. Thus, the expected value of $\sum_{j<i} 2^{|S_i \cap S_j|}$ is

$$\begin{aligned}
\mathbb{E}\left[\sum_{j<i} 2^{|S_i \cap S_j|}\right] &= \sum_{j<i} \mathbb{E}\left[2^{\sum_{k\in[\ell]} Y_{j,k}}\right] \\
&= \sum_{j<i} \mathbb{E}\left[\prod_{k\in[\ell]} 2^{Y_{j,k}}\right] \\
&= \sum_{j<i} \prod_{k\in[\ell]} \mathbb{E}\left[2^{Y_{j,k}}\right] \\
&= (i-1) \cdot (3/2)^{t-\ell} \cdot 2^{2\ell-t}\,, \qquad (6.1)
\end{aligned}$$

where the last equality is because for every $k \in [t-\ell]$ it holds that $\Pr[Y_{j,k} = 1] = 1/2$ (since $|B_k| = 2$), and for every $k \in \{t-\ell+1, ..., \ell\}$ it holds that $Y_{j,k} \equiv 1$ (since B_k is a singleton). Now, plugging-in $t = \lceil (1-4\alpha)\cdot\ell \rceil$ and $\ell = \frac{\log(\rho)}{1-\alpha}$ into Eq. (6.1), we can upper-bound the expression by $(i-1)\cdot\rho$. [16] Hence, the algorithm can find a set S_i such that $\sum_{j<i} 2^{|S_i \cap S_j|} \leq (i-1)\cdot\rho$ by trying out all $2^{t-\ell} < 2^{\ell}$ possibilities. ■

As shown in [RRV02], Trevisan's proof [Tre01] that the Nisan-Wigderson construction [NW94] yields an extractor also extends to the setting when the combinatorial design is a weak design as in Definition 6.1. Specifically:

Theorem 6.3 *(extractors from weak designs [RRV02, Prop. 10]). Let $m < k < n$ be three integers, and let $\epsilon > 0$. Let $\mathtt{ECC} : \{0,1\}^n \to \{0,1\}^{\bar{n}}$ be a code such that in every Hamming ball of radius $1/2 - \delta$ in $\{0,1\}^{\bar{n}}$ there exist at most $1/\delta^2$ codewords, where $\delta = \epsilon/4m$. Let $S_1, ..., S_m \subseteq [t]$ be an (m, ℓ, t, ρ) weak design with $\ell = \log(\bar{n})$ and $\rho = \frac{k - 3\cdot\log(m/\epsilon) - t - 3}{m}$.*

Then, the function $E : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ that is defined by $E(x,z) = (\mathtt{ECC}(x)_{z_{S_1}}, ..., \mathtt{ECC}(x)_{z_{S_m}})$ is a (k,ϵ)-extractor.

By combining Theorem 6.3 and Proposition 4.14, we obtain the following:

Corollary 6.4 *(samplers from weak designs). Let $m < k < n$ be three integers, and let $\epsilon > 0$. Let $\mathtt{ECC} : \{0,1\}^n \to \{0,1\}^{\bar{n}}$ be a code such that in every Hamming ball of radius $1/2 - \delta$ in $\{0,1\}^{\bar{n}}$ there exist at most $1/\delta^2$ codewords, where $\delta = \epsilon/4m$. Let $S_1, ..., S_m \subseteq [t]$ be an (m, ℓ, t, ρ) weak design with $\ell = \log(\bar{n})$ and $\rho = \frac{k - 3\cdot\log(m/\epsilon) - t - 3}{m}$.*

Then, the function $Samp : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ that is defined by $Samp(x,z) = (\mathtt{ECC}(x)_{z_{S_1}}, ..., \mathtt{ECC}(x)_{z_{S_m}})$ is an averaging sampler with accuracy ϵ and error 2^{k-n}.

[15] That is, for each $k \in [\ell]$ let X_k be a random element from the block B_k, such that for $k \neq k' \in [\ell]$ it holds that X_k and $X_{k'}$ are independent. Then, $S_i = \cup_{k\in[\ell]} X_k$.

[16] Denoting $c = \log(e)/2$ and $t = (1+4\beta)\cdot\ell$, where $\beta \geq \alpha$, we have that $2^{2\ell-t}\cdot(3/2)^{t-\ell} < 2^{2\ell-t}\cdot e^{(t-\ell)/2} = 2^{2\ell-t+c\cdot(t-\ell)} \leq 2^{\frac{1-4(1-c)\cdot\beta}{1-\alpha}\cdot\log(\rho)} < \rho$.

6.2 An ϵ-balanced code in sparse $\mathcal{TC}^0$

Following Corollary 6.4, our goal now is to construct a $\mathcal{TC}^0$ circuit with a super-linear number wires that computes an *error-correcting code* that is list-decodable up to distance $1/2 - \delta$ with list size $\text{poly}(1/\delta)$ and rate $\text{poly}(1/\delta)$. We will do this by constructing a code with distance $1/2 - \epsilon$, where $\epsilon = \delta^2$, and then relying on the Johnson bound. In fact, we will actually construct an ϵ-balanced code (i.e., a linear code such that all codewords have relative Hamming weight $1/2 \pm \epsilon$).

As described in the introduction, the construction will consist of two parts. We will first construct a code with constant relative distance, and then show how to amplify the distance from $\Omega(1)$ to $1/2 - \epsilon$.

Proposition 6.5 *(a code with constant relative distance in sparse $\mathcal{TC}^0$). There exists a polynomial-time algorithm that is given as input 1^n and a constant $d \in \mathbb{N}$, and outputs a $\mathcal{TC}^0$ circuit C that satisfies the following:*

1. *The circuit C maps n input bits to $\hat{n} = O(n)$ input bits.*
2. *For every $x \in \{0,1\}^n$ such that $x \neq 0^n$, the relative Hamming weight of $C(x)$ is at least 3^{-d}.*
3. *Each output bit of C is a linear function of the input bits.*
4. *The circuit C has depth $2d$ and $n^{1+O(1/d)}$ wires.*

Proof. Assume that n is of the form r^d, for $r \in \mathbb{N}$ (if necessary, pad the input with zeroes such that the input length will be a power of 2^d). Fix a linear code $\texttt{ECC}$ that maps strings of length r to strings of length $\bar{r} = O(r)$ and has relative distance at least $1/3$ (e.g., we can use the ϵ-balanced codes of [NN93, TS17]).

Let $x \in \{0,1\}^n$ be an input for the circuit C. We think of x as a tensor $M^{(0)}$ of dimensions $[r]^d$; that is, for every $\vec{t} \in [r]^d$, the $\vec{t}^{th}$ entry of $M^{(0)}$ is denoted by $M^{(0)}_{\vec{t}} \in \{0,1\}$. The circuit C will iterative compute a sequence $M^{(1)}, ..., M^{(d)}$ of tensors, and the message $x = M^{(0)}$ will be mapped to the final codeword $\hat{x} = M^{(d)}$.

For each $i \in [d]$, the tensor $M^{(i)}$ is defined as follows. The dimensions of $M^{(i)}$ are $[\bar{r}]^i \times [r]^{d-i}$. For every pair $(\vec{t_{\leq i-1}}, \vec{t_{\geq i+1}}) \in [\bar{r}]^{i-1} \times [r]^{d-i}$, we denote by $M^{(i-1)}_{\vec{t}_{\leq i-1},\star,\vec{t}_{\geq i+1}}$ the r-bit vector $M^{(i-1)}_{\vec{t}_{\leq i-1},\star,\vec{t}_{\geq i+1}} \stackrel{\text{def}}{=\!=} M^{(i-1)}_{(\vec{t}_{\leq i-1},1,\vec{t}_{\geq i+1})}, ..., M^{(i-1)}_{(\vec{t}_{\leq i-1},m,\vec{t}_{\geq i+1})} \in \{0,1\}^r$. Then, for every $\vec{t} \in [\bar{r}]^i \times [r]^{d-i}$, we think of $\vec{t}$ as a triplet $\vec{t} = (\vec{t}_{\leq i-1}, u, \vec{t}_{\geq i+1}) \in [\bar{r}]^{i-1} \times [\bar{r}] \times [r]^{d-i}$, and define $M^{(i)}_{\vec{t}} = \left(\texttt{ECC}\left(M^{(i-1)}_{\vec{x}_{\leq i-1},\star,\vec{x}_{\geq i+1}}\right)\right)_v$ (i.e., $M^{(i)}_{(\vec{t}_{\leq i-1},v,\vec{t}_{\geq i+1})}$ is the v^{th} coordinate of the encoding of $M^{(i-1)}_{\vec{t}_{\leq i-1},\star,\vec{t}_{\geq i+1}}$ by $\texttt{ECC}$).

The final codeword $\hat{x} = M^{(d)}$ is of dimensions $[\bar{r}]^d$, which means that it represents a string of length $\hat{n} = (O(r))^d = O(n)$. The fact that every non-zero message $x \in \{0,1\}^n$ is mapped to a codeword $\hat{x} \in \{0,1\}^{\hat{n}}$ with relative Hamming weight at least $(1/3)^d$ follows from the properties of $\texttt{ECC}$ and from well-known properties of tensor codes; for completeness, we include a proof in Appendix B. Also note that each bit of $\hat{x}$ is indeed a linear function of x, because $\texttt{ECC}$ is linear (which means that in each iteration $i \in [d]$, every bit of $M^{(i)}$ is a linear function of $M^{(i-1)}$).

Finally, let us fix $i \in [d]$, and describe how to compute $M^{(i)}$ from $M^{(i-1)}$ in depth two with $O(n \cdot r^2)$ wires. Since $\texttt{ECC}$ is linear, for each $\vec{t} = (\vec{t}_{\leq i-1}, v, \vec{t}_{\geq i+1}) \in [\bar{r}]^{i-1} \times [\bar{r}] \times [r]^{d-i}$ it

holds that $M^{(i)}_{\vec{t}} = \mathtt{ECC}\left(M^{(i-1)}_{\vec{t}_{\leq i-1},\star,\vec{t}_{\geq i+1}}\right)_v$ is a linear function of the r-bit string $M^{(i-1)}_{\vec{t}_1,\star,\vec{t}_2} \in \{0,1\}^r$. Thus, each entry of $M^{(i)}$ can be computed from $M^{(i-1)}$ by a depth-2 $\mathcal{TC}^0$ circuit with $O(r^2)$ wires (see, e.g., [PS94, Sec. 3]), which means that $M^{(i)}$ can be computed from $M^{(i-1)}$ by a depth-2 $\mathcal{TC}^0$ circuit with $O(n \cdot r^2)$ wires. Overall, the final circuit C is of depth $2d$ (since it is comprised of d circuits of depth two), and the number of wires in C is at most $O(n \cdot r^2) < n^{1+O(1/d)}$. ■

We now show how to amplify the distance of the code from Proposition 6.5 from $\Omega(1)$ to $1/2 - \epsilon$.

Proposition 6.6 *(amplifying the distance of the code from Proposition 6.5). There exists a polynomial-time algorithm that is given as input $1^{\hat{n}}$, a constant $\rho > 0$, and $\epsilon = \epsilon(\hat{n}) > 0$, and outputs a $\mathcal{TC}^0$ circuit C such that:*

1. *The circuit C maps $\hat{n}$ input bits to $\bar{n} = \hat{n} \cdot (1/\epsilon)^{O(1/\rho)}$ output bits.*
2. *For every $\hat{x} \in \{0,1\}^{\hat{n}}$ with relative Hamming weight at least ρ, the relative Hamming weight of $\bar{x} = C(\hat{x})$ is between $1/2 - \epsilon$ and $1/2$.*
3. *Each output bit of C is a linear function of the input bits.*
4. *The circuit C has depth two and $\hat{n} \cdot (1/\epsilon)^{O(1/\rho)}$ wires.*

Proof. The algorithm first constructs an expander graph G on $\hat{n}$ vertices; that is, a d_G-regular graph over the vertex-set $[\hat{n}]$ vertices with constant spectral gap. [17] Consider a random walk that starts from a uniform $i \in [\hat{n}]$ and walks $\ell - 1$ steps, where $\ell = \frac{c_G}{\rho} \cdot \log(1/\epsilon)$ and c_G is a sufficiently large constant that depends only on G. By the hitting property of expander random walks (see, e.g., [Gol08, Thm 8.28]), with probability at least $1 - \epsilon$ such a walk hits $i \in [\hat{n}]$ such that $x_i \neq 0$ (this is because the set $\{i \in [\hat{n}] : x_i \neq 0\}$ has density at least ρ). Thus, if we first take such a random walk, and then output a random parity of the values of $\hat{x}$ at the coordinates corresponding to the vertices in the walk, the output will equal one with probability at least $1/2 - \epsilon$ and at most $1/2$.

The mapping of $\hat{x}$ to $\bar{x} = C(\hat{x})$ is obtained by considering all the possible outcomes of the random process above. Specifically, for every random walk $W = \left(i^{(W)}_1, ..., i^{(W)}_\ell\right)$ of length $\ell - 1$ on G, and every subset $S \subseteq [\ell]$, we have a corresponding coordinate (W, S) in $C(\hat{x})$. The value of $C(\hat{x})$ at coordinate (W, S) is the parity of the bits of $\hat{x}$ in the locations corresponding to S in walk W; that is, $C(\hat{x})_{(W,S)} = \oplus_{j \in S} \hat{x}_{i^{(W)}_j}$.

Note that the length of $C(\hat{x})$ is $\hat{n} \cdot (d_G)^{\ell-1} \cdot 2^\ell = \hat{n} \cdot (1/\epsilon)^{c'_G/\rho}$, where c'_G is a large constant that only depends on G. Also, the mapping of $\hat{x}$ to $C(\hat{x})$ is linear, and moreover every coordinate of $C(\hat{x})$ is the parity of ℓ coordinates of $\hat{x}$. Thus, $C(\hat{x})$ can be computed by a $\mathcal{TC}^0$ circuit of depth two using at most $\hat{n} \cdot (1/\epsilon)^{c/\rho} \cdot \ell^2 < \hat{n} \cdot (1/\epsilon)^{2c/\rho}$ wires. ■

By combining Propositions 6.5 and 6.6 we obtain the following:

Proposition 6.7 *(an ϵ-balanced code in sparse $\mathcal{TC}^0$). There exists a polynomial-time algorithm that gets inputs 1^n and $\epsilon = \epsilon(n)$ and a constant $d \in \mathbb{N}$, and outputs a $\mathcal{TC}^0$ circuit such that:*

[17] For a suitable construction see, e.g., [Gol08, Thm E.10]. This specific construction requires $\hat{n}$ to be a square, so we might need to pad the input $x \in \{0,1\}^{\hat{n}}$ with zeroes such that it will be of length $4^k = (2^k)^2$, for $k \in \mathbb{N}$. Since such a padding will not affect the rest of the argument, we ignore this issue.

1. *The circuit computes a linear code that maps messages of length n to codewords of length $\bar{n} = n \cdot (1/\epsilon)^{O(3^d)}$ such that every codeword has relative Hamming weight $1/2 \pm \epsilon$.*

2. *The circuit has depth $2d$ and $n^{1+O(1/d)} + n \cdot (1/\epsilon)^{O(3^d)}$ wires.*

Relying on the Johnson bound, we obtain the list-decodable code that is needed for Corollary 6.4 as a corollary of Proposition 6.7:

Corollary 6.8 *(a list-decodable code in sparse $\mathcal{TC}^0$). There exists a polynomial-time algorithm that gets inputs 1^n and $\delta = \delta(n)$ and a constant $d \in \mathbb{N}$, and outputs a $\mathcal{TC}^0$ circuit such that:*

1. *The circuit computes a linear code mapping messages of length n to codewords of length $\bar{n} = n \cdot (1/\delta)^{O(3^d)}$ such that in any Hamming ball of radius $1/2 - \delta$ in $\{0,1\}^{\bar{n}}$ there exist at most $O(1/\delta^2)$ codewords.*

2. *The circuit has depth $2d$ and $n^{1+O(1/d)} + n \cdot (1/\delta)^{O(3^d)}$ wires.*

Proof. We invoke Proposition 6.7 with $\epsilon = \delta^2$. The code that the circuit computes has distance $1/2 - \delta^2$. Relying on the Johnson bound (see, e.g., [AB09, Thm 19.23]), in such a code every Hamming ball of radius δ contains at most $1/\delta^2$ codewords. ■

6.3 An averaging sampler in sparse $\mathcal{TC}^0$

We now combine Lemma 6.2, Corollary 6.4, and Corollary 6.8, to get an averaging sampler that can be computed by a $\mathcal{TC}^0$ circuit with a super-linear number of wires. The sampler will get an input of length n, and for two constants $0 < \gamma \ll \beta < 1$, the sampler will output $m = n^\gamma$ bits and will have accuracy $1/m$ and error $2^{n^\beta - n}$.

Proposition 6.9 *(an averaging sampler in sparse $\mathcal{TC}^0$). There exists a polynomial-time algorithm that gets as input 1^n and three constants $d \in \mathbb{N}$ and $\gamma \leq \frac{1}{c \cdot d \cdot 3^d}$ (where $c > 1$ is some universal constant) and $\beta \geq 4/5$, and outputs a $\mathcal{TC}^0$ circuit C that satisfies the following:*

1. *The circuit C gets input $x \in \{0,1\}^n$ and outputs $2^t < n^{(1+O(1/d))\cdot(5-4\beta)}$ strings of length $m = n^\gamma$.*

2. *The function $Samp : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ such that $Samp(x,i) = C(x)_i$ (i.e., $Samp(x,i) \in \{0,1\}^m$ is the i^{th} output string of $C(x)$) is an averaging sampler with accuracy $\epsilon = 1/m$ and error $2^{n^\beta - n}$.*

3. *The depth of C is $2d+1$ and its number of wires is at most $n^{(1+O(1/d))\cdot(5-4\beta)}$.*

In particular, if $\beta \geq 1 - 1/5d$, then both the number of outputs of C (i.e., 2^t) and the number of wires in C are less than $n^{1+O(1/d)}$.

Proof. We first use Corollary 6.8 with the parameter value $\delta = \epsilon/4m$ to construct a circuit C_0 of depth $2d$ that encodes its input $x \in \{0,1\}^n$ to a codeword $\bar{x}$ of length $\bar{n}$. Then, we use Lemma 6.2 to construct an (m, ℓ, t, ρ) weak design $S_1, ..., S_m \subseteq [t]$ with the following parameters: For $\alpha = 1 - \beta + (c \cdot 3^{d+1}) \cdot \gamma < 1/4$ (the inequality is since $\beta > 4/5$ and γ is sufficiently small), we construct a design with $\ell = \log(\bar{n})$ and $\rho = 2^{(1-\alpha)\cdot\ell}$ and $t = \lceil (1+4\alpha) \cdot \ell \rceil$. Now, define a function $Samp : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ as in Corollary 6.4; that is, for $x \in \{0,1\}^n$ and $z \in \{0,1\}^t$, the m-bit string $Samp(x,z)$ is the projection

of $\bar{x}$ to the coordinates $z_{S_1}, ..., z_{S_m}$. The circuit C outputs the 2^t strings corresponding to $\{Samp(x,z)\}_{z\in\{0,1\}^t}$, where each output string is a projections of m bits of $\bar{x}$.

Let $k = n^\beta$. An elementary calculation shows that $\rho = 2^{(1-\alpha)\cdot\ell} < \frac{k-3\cdot\log(m/\epsilon)-t-3}{m}$. [18] Thus, relying on Corollary 6.4, the function $Samp$ is an averaging sampler with accuracy ϵ and error 2^{k-n}. The depth of C is $2d+1$ (since the depth of C_0 is $2d$, and the 2^t outputs are projections of $\bar{x}$). Finally, the number of wires in C_0 is $n^{1+O(1/d)} + n \cdot (m/\epsilon)^{O(3^d)} < n^{1+O(1/d)}$, and the number of wires between $\bar{x}$ and the outputs is $2^t \cdot m = 2^{\lceil(1+4\alpha)\cdot\log(\bar{n})\rceil} \cdot m = n^{(1+O(\gamma\cdot 3^d))(1+4\alpha)} = n^{(1+O(1/d))\cdot(5-4\beta)}$. ■

6.4 Proof of Theorem 1.2

Let us now formally state Theorem 1.2 and prove it using the averaging sampler from Proposition 6.9. Towards stating the theorem, for any $n, d, k \in \mathbb{N}$, denote by $\mathcal{C}_{n,d,n^k}$ either the class of linear threshold circuits over n input bits of depth d and with at most n^k wires.

Theorem 6.10 *(Theorem 1.2, restated). Assume that for every $d \in \mathbb{N}$ and for some $\beta = \beta_d \geq 4/5$ there exists an algorithm that gets as input a circuit $C' \in \mathcal{C}_{n,d,n^{(1+O(1/d))\cdot(5-4\beta)}}$, runs in time $T(n)$, and satisfies the following: If C' rejects all but at most 2^{n^β} of its inputs, then the algorithm rejects C', and if C' accepts all but at most 2^{n^β} of its inputs, then the algorithm accepts C'.*

Then, there exists an algorithm that for every $k \in \mathbb{N}$ and $d \in \mathbb{N}$, when given as input a circuit $C \in \mathcal{C}_{m,d,m^k}$, runs in time $T(m^{O(k\cdot d\cdot 3^d)})$ (where the O-notation hides some fixed universal constant), and satisfies the following: If C accepts at least $2/3$ of its inputs then the algorithm accepts C, and if C rejects at least $2/3$ of its inputs then the algorithm rejects C.

To obtain the parameters of Theorem 1.2, use the value $\beta_d = 1 - 1/5d$, in which case the number of wires of C' is $n^{1+O(1/d)}$; and for every $k \in \mathbb{N}$, we can assume that d is sufficiently large such that $O(k \cdot d \cdot 3^d \cdot 4^{-d}) < 1$, in which case the running time of the algorithm is at most $T(m^{O(k\cdot d\cdot 3^d)}) = 2^{m^{1-\Omega(1)}}$ (due to the hypothesis that $T(n) = 2^{n^{1/4^d}}$).

Proof of Theorem 6.10. Let $C \in \mathcal{C}_{m,d,m^k}$ be an input to the algorithm, let $\gamma = 1/c \cdot k \cdot d \cdot 3^d$ for a sufficiently large universal constant $c > 1$, and let $\beta = \beta_{3d+2}$. We will construct a circuit $C' \in \mathcal{C}_{n,3d+2,n^{(1+O(1/d))\cdot(5-4\beta)}}$, where $n = m^{1/\gamma}$, such that the following holds: If C rejects at least a $2/3$ fraction of its inputs, then C' rejects all but at most 2^{n^β} inputs; and if C accepts at least a $2/3$ fraction of its inputs, then C' accepts all but 2^{n^β} of its inputs. Then, we can use the quantified derandomization algorithm for C', which runs in time $T(n) = T\left(m^{c\cdot k\cdot d\cdot 3^d}\right)$, to decide whether the acceptance probability of C is at least $2/3$ or at most $1/3$.

To construct C', we first use Proposition 6.9 to construct a $\mathcal{TC}^0$ circuit $Samp : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ that is an averaging sampler with the following properties: The input length is n, the output length is $m = n^\gamma$, the accuracy is $\epsilon = n^{\Omega(1)} < 1/100$, and the error is $\delta = 2^{n^\beta - n}$; the number of wires in $Samp$ is at most $n^{(1+O(1/d))\cdot(5-4\beta)}$, and its depth is $2d+1$. The circuit C' first computes the sampler $Samp$, then evalutes C in parallel on each of the $2^t < n^{(1+O(1/d))\cdot(5-4\beta)}$ outputs of the sampler, and finally computes the majority of

[18] To see that this holds, let $c' > 1$ be the universal constant such that $\bar{n} \leq n \cdot (m/\epsilon)^{c'\cdot 3^d}$. Then, note that $\alpha = 1 - \beta + (c'\cdot 3^{d+1})\cdot\gamma > \frac{1-\beta+(2c'\cdot 3^d+1)\cdot\gamma}{1+2c'\cdot\gamma\cdot 3^d} = 1 - \frac{\beta-\gamma}{1+2c'\cdot\gamma\cdot 3^d}$. It follows that $\log(\rho) = (1-\alpha)\cdot\ell < \log(k/2m)$, since $1-\alpha < \frac{\beta-\gamma-1/\log(n)}{1+2c'\cdot\gamma\cdot 3^d}$. We can thus deduce that $\rho \leq k/2m < \frac{k-3\cdot\log(m/\epsilon)-t-3}{m}$.

the 2^t evaluations of C. That is, $C'(x) = MAJ_{z\in\{0,1\}^t}[C(Samp(x,z))]$. The circuit C' is of depth $(2d+1)+d+1 = 3d+2$, and its number of wires is at most $n^{(1+O(1/d))\cdot(5-4\beta)} + m^k = n^{(1+O(1/d))\cdot(5-4\beta)}$, where we relied on the fact that $m^k < n$.

Note that for any $x \in \{0,1\}^n$ such that $\Pr_{z\in\{0,1\}^t}[C(Samp(x,z)) = 1] \in \Pr[C(\mathbf{u}_n) = 1] \pm \epsilon$, we have that $C'(x)$ outputs the most frequent value of C. Since the accuracy of the sampler is $2^{n^\beta - n}$, the number of strings in $\{0,1\}^n$ such that $\Pr_{z\in\{0,1\}^t}[C(Samp(x,z)) = 1] \notin \Pr[C(\mathbf{u}_n) = 1] \pm \epsilon$ is at most 2^{n^β}. Thus, the number of strings $x \in \{0,1\}^n$ such that $C'(x)$ does not output the most frequent value of C is at most 2^{n^β}. ■

Observe that the circuit C' that we constructed in the proof of Theorem 6.10 consists of the sampler from Proposition 6.9, which only uses majority gates; of copies of the initial circuit C; and of an additional majority gate. Thus, the statement of Theorem 6.10 holds even if we interpret $\mathcal{C}_{n,d,w}$ as the class of circuits with majority gates (rather than linear threshold circuits) over n input bits of depth d and with at most w wires.

7 Quantified derandomization of depth-2 linear threshold circuits

In this section we construct a quantified derandomization algorithm for depth-2 linear threshold circuits with $n^{3/2-\Omega(1)}$ wires. In fact, we construct a *pseudorandom generator* for the class of depth-2 linear threshold circuits with $n^{3/2-\Omega(1)}$ wires that either accept all but $B(n) = 2^{n^{\Omega(1)}}$ of their inputs or reject all but $B(n)$ of their inputs. That is, we construct an algorithm G that gets as input a seed s of length $\tilde{O}(\log(n))$, and outputs an n-bit string such that for every $C \in \mathcal{C}_{n,2,n^{3/2-\Omega(1)}}$ the following holds: If C accepts all but $B(n) = 2^{n^{\Omega(1)}}$ of its inputs, then the probability that $C(G(s)) = 1$ is very high, and if C rejects all but $B(n)$ of its inputs, then the probability that $C(G(s)) = 0$ is very low.

The pseudorandom generator that we construct in this appendix is incomparable to the pseudorandom generator of Servedio and Tan [ST17b]. On the one hand, their generator is $\frac{1}{\mathrm{poly}(n)}$-pseudorandom for *every* depth-two linear threshold circuit, whereas our generator only "fools" circuits with acceptance probability that is either very high or very low. Moreover, their generator can handle circuits with $n^{2-\Omega(1)}$ wires, whereas our generator can only handle circuits with $n^{3/2-\Omega(1)}$ wires. But on the other hand, their generator requires a seed of length $n^{1-\Omega(1)}$, whereas our generator only requires a seed of length $\tilde{O}(\log(n))$.

Recall that our main quantified derandomization algorithm (from Theorem 1.1) leverages the techniques underlying the correlation bounds of Chen, Santhanam, and Srinivasan [CSS16] for depth-d linear threshold circuits. The generator in this section leverages the techniques underlying the correlation bounds of Kane and Williams [KW16] for depth-2 linear threshold circuits.

Specifically, our first step is to prove a derandomized version of the restriction lemma of Kane and Williams [KW16]. We actually state a slightly generalized version, which is implicit in the original argument. We say that a distribution $\mathbf{y}$ over $\{0,1\}^n$ is p-bounded in pairs if for every $i \neq j \in [n]$ it holds that $\Pr[\mathbf{y}_i = 1] \leq p$ and $\Pr[\mathbf{y}_i = 1 \wedge \mathbf{y}_j = 1] \leq p^2$. One example for a distribution that is p-bounded in pairs is the distribution $\mathbf{y}$ in which each coordinate is independently set to 1 with probability p. Another example, which is used in [KW16], is the following: Consider a equipartition of $[n]$ to $p \cdot n$ disjoint sets $S_1, ..., S_{p\cdot n}$; then, sampling $y \sim \mathbf{y}$ is equivalent to uniformly choosing a single coordinate

in each set S_i in the partition, fixing y in the chosen coordinates to one, and fixing y in all other coordinates to zero (so that the Hamming weight of $y \sim \mathbf{y}$ is always $p \cdot n$).

Proposition 7.1 *(derandomized version of [KW16, Lem. 3.1]). Let $\Phi = (w, \theta)$ be an LTF on m input bits. For $p > 0$, let $\mathbf{y}$ be a distribution over $\{0,1\}^n$ that is p-bounded in pairs, and let $\mathbf{z}$ be a distribution over $\{-1,1\}^n$ that is $\frac{1}{\mathrm{poly}(m)}$-pseudorandomly concentrated. Let $\boldsymbol{\rho}$ be the distribution over restrictions obtained by sampling $y \sim \mathbf{y}$ in order to determine which variables are kept alive (the i^{th} variable is kept alive if and only if $\mathbf{y}_i = 1$), and independently sampling $z \sim \mathbf{z}$ to determine values for the fixed variables. Then,*

$$\Pr_{\rho \sim \boldsymbol{\rho}}[\Phi\restriction_\rho \text{ depends on more than one input bit}] = O(m \cdot p^{3/2}) \,.$$

Proof. For every choice of $y \sim \mathbf{y}$, let $I = I_y \subseteq [n]$ be the set of live variables (i.e., $I = \{i \in [n] : y_i = 1\}$). Then, the probability that $\Phi\restriction_\rho$ depends on more than one input bit is at most

$$\begin{aligned}
&\Pr_{\rho \sim \boldsymbol{\rho}}\left[|I| \geq 2 \wedge \Phi\restriction_\rho \text{ is not constant}\right] \\
&= \mathbb{E}_{y \sim \mathbf{y}}\left[\Pr_{z \sim \mathbf{z}}\left[|I| \geq 2 \wedge \Phi\restriction_\rho \text{ is not constant}\right]\right] \\
&= \mathbb{E}_{y \sim \mathbf{y}}\left[\mathbf{1}_{|I| \geq 2} \cdot \Pr_{z \sim \mathbf{z}}\left[\Phi\restriction_\rho \text{ is not constant}\right]\right] \,,
\end{aligned} \tag{7.1}$$

where the first equality relied on the fact that y and z are sampled independently, and the second equality is since the random variable I only depends on y (and not on z).

Fix an arbitrary choice of y, and let us upper-bound the probability over $z \sim \mathbf{z}$ that $\Phi\restriction_\rho$ is not constant. Note that $\Phi\restriction_\rho$ is a constant function if and only if

$$\left|\theta - \left\langle w_{[m]\setminus I}, z_{[m]\setminus I}\right\rangle\right| > \|w_I\|_1 \iff \left\langle w_{[m]\setminus I}, z_{[m]\setminus I}\right\rangle \notin \theta \pm \|w_I\|_1 \,. \tag{7.2}$$

For each $i \in [m]$, let k_i be the index of the i^{th} variable when the variables are sorted according to the magnitudes $|w_i|$ in ascending order (breaking ties arbitrarily). In [KW16, Proof of Lemma 1.1] it is shown that the probability over a *uniform choice of* z that Eq. (7.2) holds is at most $\sum_{i \in I} \frac{O(1)}{\sqrt{k_i}}$. Since $\mathbf{z}$ is $(1/\mathrm{poly}(m))$-pseudorandomly concentrated, the probability under $z \sim \mathbf{z}$ that Eq. (7.2) holds is at most $\sum_{i \in I} \frac{O(1)}{\sqrt{k_i}} + \frac{1}{\mathrm{poly}(m)}$. Therefore, the expression in Eq. (7.1) is upper-bounded by

$$\begin{aligned}
&\mathbb{E}_{y \sim \mathbf{y}}\left[\mathbf{1}_{|I| \geq 2} \cdot \sum_{i \in I} \frac{O(1)}{\sqrt{k_i}}\right] + \frac{1}{\mathrm{poly}(m)} \\
&= \mathbb{E}_{y \sim \mathbf{y}}\left[\sum_{i \in [m]} \frac{O(1)}{\sqrt{k_i}} \cdot \mathbf{1}_{i \in I \wedge |I| \geq 2}\right] + \frac{1}{\mathrm{poly}(m)} \\
&= \sum_{i \in [m]} \frac{O(1)}{\sqrt{k_i}} \cdot \Pr_{y \sim \mathbf{y}}[i \in I \wedge |I| \geq 2] + \frac{1}{\mathrm{poly}(m)} \,.
\end{aligned} \tag{7.3}$$

For any fixed $i \in [m]$, we upper-bound the probability of the event $i \in I \wedge |I| \geq 2$ in two ways: The first upper-bound is $\Pr[i \in I] \leq p$, and the second upper-bound is $\Pr[\exists j \in [m] \setminus \{i\}, j \in I \wedge i \in I] < m \cdot p^2$ (since $\mathbf{y}$ is p-bounded in pairs). Hence,

$$\Pr_{y \sim \mathbf{y}}[i \in I \wedge |I| \geq 2] \leq \min\left\{p, m \cdot p^2\right\} \leq \sqrt{m \cdot p^3} \,,$$

which implies that the expression in Eq. (7.3) is upper-bounded by

$$\sqrt{m \cdot p^3} \cdot \sum_{i \in [m]} \frac{O(1)}{\sqrt{k_i}} + \frac{1}{\mathrm{poly}(m)} = O\left(\sqrt{m} \cdot p^{3/2} \cdot \sum_{i \in [m]} \frac{1}{\sqrt{i}}\right) = O\left(m \cdot p^{3/2}\right) . \qquad \blacksquare$$

Our pseudorandom generator, which is contructed next, is based on an application of Proposition 7.1 as well as on the pseudorandom generator of Gopalan, Kane, and Meka (i.e., Theorem 4.8).

Theorem 7.2 *(quantified derandomization of depth-2 linear threshold circuits with $n^{3/2-\Omega(1)}$ wires). There exists a polynomial-time algorithm G that is given as input a random seed s of length $\tilde{O}(\log(n))$ and a constant $\epsilon > 0$, and outputs a string $G(s,\epsilon) \in \{0,1\}^n$ such that for every $C \in \mathcal{C}_{n,2,n^{3/2-\epsilon}}$ the following holds:*

1. *If C accepts all but at most $B(n) = 2^{n^{\epsilon/2}}$ inputs, then $\Pr_s[C(G(s,\epsilon)) = 1] = 1 - o(1)$.*
2. *If C rejects all but at most $B(n)$ inputs, then $\Pr_s[C(G(s,\epsilon)) = 1] = o(1)$.*

Proof. Let $\delta \in (\epsilon/2, 2\epsilon/3)$ such that $p = n^{-(1-\delta)}$ is a power of two. The algorithm first samples a restriction that meets the requirements of Proposition 7.1, as follows: The distribution $\mathbf{y}$ over $\{0,1\}^n$ is obtained by sampling a string y' from a distribution over $\{0,1\}^{\log(1/p)\cdot n}$ that is $\frac{1}{\mathrm{poly}(n)}$-almost $O(\log(n))$-wise independent, and setting $\mathbf{y}_i = 1$ if and only if the i^{th} block in y' is all zeroes; and the distribution $\mathbf{z}$ is $\frac{1}{\mathrm{poly}(n)}$-pseudorandomly concentrated. The required seed length to sample such a restriction is dominated by the seed length required to sample $z \sim \mathbf{z}$, which (using Theorem 4.8) is $O(\log(n) \cdot (\log\log(n))^2)$.

We say that a restriction ρ is successful if the circuit $C\restriction_\rho$ can be computed by a single LTF, and if at least $\frac{1}{2} \cdot (p \cdot n) = \frac{1}{2} \cdot n^\delta$ variables remain alive under ρ. We first claim that the probability that ρ is successful is $1 - o(1)$. According to Fact 4.9, with probability $1 - 1/\mathrm{poly}(n)$ at least $\frac{1}{2} \cdot n^\delta$ variables remain alive under ρ. To see that with high probability $C\restriction_\rho$ can be computed by a single LTF, let $\mathcal{G}$ be the set of gates in the bottom layer of C. We say that a gate Φ is non-trivial if Φ depends on more than a single input bit; note that any trivial gate can be replaced by a constant or by an input bit (or its negation). Then, the expected number of non-trivial gates in the bottom layer of $C\restriction_\rho$ is

$$\begin{aligned}
\mathbb{E}_\rho\left[\sum_{\Phi \in \mathcal{G}} \mathbf{1}_{\Phi\restriction_\rho \text{ is non-trivial}}\right] &= \sum_{\Phi \in \mathcal{G}} \Pr_\rho[\Phi\restriction_\rho \text{ is non-trivial}] \\
&= O\left(\sum_{\Phi \in \mathcal{G}} \text{fan-in}(\Phi) \cdot p^{3/2}\right) \\
&= O\left(n^{3/2-\epsilon} \cdot n^{3\delta/2-3/2}\right) ,
\end{aligned}$$

which is $o(1)$, since $\delta < 2\epsilon/3$. Therefore, the probability that there are no non-trivial gates in the bottom layer of $C\restriction_\rho$ is $1 - o(1)$.

After sampling the restriction ρ, the algorithm samples a string $x \in \{0,1\}^{|\rho^{-1}(\star)|}$ using the pseudorandom generator G' for LTFs from Theorem 4.8, instantiated with error parameter $1/\mathrm{poly}(n)$, and outputs the n-bit string that is obtained by completing x to an n-bit string according to ρ.

To see that the algorithm is correct, assume that C accepts all but $2^{n^{\epsilon/2}}$ of its inputs. Then, for every successful restriction ρ, the acceptance probability of $C\restriction_\rho$ is $1 - o(1)$ (since ρ keeps at least $\frac{1}{2} \cdot n^\delta = \omega(n^{\epsilon/2})$ variables alive). Thus,

$$\begin{aligned}\Pr_s[C(G(s,\epsilon)) = 0] &\leq \Pr_\rho[\rho \text{ not successful}] + \Pr_s[C(G(s,\epsilon)) = 0 | \rho \text{ successful}] \\ &\leq o(1) + \max_{\rho \text{ successful}} \Pr_{s'}[C\restriction_\rho(G'(s')) = 0] \,,\end{aligned}$$

which is $o(1)$ since G' is $\frac{1}{\text{poly}(n)}$-pseudorandom for LTFs. Similarly, if C rejects all but $2^{n^{\epsilon/2}}$ of its inputs, then $\Pr[C(G(s)) = 1] = o(1)$. ■

8 Restrictions for sparse $\mathcal{TC}^0$ circuits: A potential path towards $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$

Recall that the best currently-known lower bounds for $\mathcal{TC}^0$ circuits of arbitrary constant depth d are for circuits with $n^{1+\exp(-d)}$ wires. We now present an open problem that involves restrictions for $\mathcal{TC}^0$ circuits with only $n^{1+O(1/d)}$ wires, and show that a resolution of this open problem would imply that $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$.

Towards presenting the problem, fix some class $\mathcal{C}_{simple}$ of "simple" functions such that the following holds: There exists a deterministic algorithm that gets as input $C' \in \mathcal{C}_{simple}$, runs in sufficiently small sub-exponential time, and distinguishes between the case that the acceptance probability of C' is at least $2/3$ and the case that the acceptance probability of C' is at most $1/3$. Then, the problem is the following:

Open Problem 1 *(deterministic restriction algorithm for sparse $\mathcal{TC}^0$ circuits). Construct a deterministic algorithm that gets as input a $\mathcal{TC}^0$ circuit $C : \{-1,1\}^n \to \{-1,1\}$ of depth d with $n^{1+O(1/d)}$ wires, runs in time at most $2^{n^{1/4^d}}$, and finds a set $S \subseteq \{-1,1\}^n$ and $C' \in \mathcal{C}_{simple}$ such that $|S| \geq 10 \cdot 2^{n^{1-1/5d}}$ and $C\restriction_S$ is $(1/10)$-close to C'.*

A resolution of Open Problem 1 would imply that there exists an algorithm for quantified derandomization of $\mathcal{TC}^0$ circuits of depth d with $n^{1+O(1/d)}$ wires and $B(n) = 2^{n^{1-1/5d}}$ exceptional inputs that runs in sufficiently small sub-exponential time (i.e., in time $2^{n^{1/4^d}}$). This is the case because a quantified derandomization algorithm can act similarly to our algorithm from the proof of Theorem 1.1, as follows: First find a set S such that $|S| \geq 10 \cdot 2^{n^{1-1/5d}}$ and $C\restriction_S$ is $(1/10)$-close to some $C' \in \mathcal{C}_{simple}$; then, note that $C\restriction_S$ has either very high acceptance probability or very low acceptance probability (because C has at most $B(n) \leq |S|/10$ exceptional inputs); and finally, estimate the acceptance probability of $C\restriction_S$ (by estimating the acceptance probability of C') in order to decide whether C accepts all but at most $B(n)$ of its inputs or rejects all but at most $B(n)$ of its inputs. Thus, relying on Corollary 1.3, a resolution of Open Problem 1 would imply that $\mathcal{NEXP} \not\subseteq \mathcal{TC}^0$.

Acknowledgements

This work was initiated and partially conducted while the author was visiting Rocco Servedio at Columbia, and under Rocco's guidance. The author is very grateful to Rocco, who

declined co-authorship of the paper, for his guidance, for many useful ideas, and for numerous inspiring conversations. The author thanks his advisor, Oded Goldreich, for the very useful idea to use tensor codes in the proof of Theorem 1.2, and for his guidance and support during the research and writing process. The author also thanks Amnon Ta-Shma for a useful conversation about constructing extractors in $\mathcal{TC}^0$.

This research was partially supported by the Minerva Foundation with funds from the Federal German Ministry for Education and Research. The research was also supported by the Prof. Rahamimoff Travel Grant for Young Scientists of the US-Israel Binational Science Foundation (BSF).

References

[Aar17] Scott Aaronson. $P \stackrel{?}{=} NP$, 2017. Accessed at http://www.scottaaronson.com/papers/pnp.pdf, June 20, 2017.

[AB09] Sanjeev Arora and Boaz Barak. *Computational complexity: A modern approach*. Cambridge University Press, Cambridge, 2009.

[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 Transactions on Information Theory*, 38(2):509–516, 1992.

[AS15] Kazuyuki Amano and Atsushi Saito. A nonuniform circuit class with multilayer of threshold gates having super quasi polynomial size lower bounds against NEXP. In *Proc. 9th International Conference on Language and Automata Theory and Applications (LATA)*, pages 461–472. 2015.

[BBL92] Paul Beame, Erik Brisson, and Richard Ladner. The complexity of computing symmetric functions using threshold circuits. *Theoretical Computer Science*, 100(1):253–265, 1992.

[BIS12] Paul Beame, Russell Impagliazzo, and Srikanth Srinivasan. Approximating AC^0 by small height decision trees and a deterministic algorithm for $\#\mathrm{AC}^0SAT$. In *Proc. 27th Annual IEEE Conference on Computational Complexity (CCC)*, pages 117–125. 2012.

[BM84] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM Journal of Computing*, 13(4):850–864, 1984.

[Bra10] Mark Braverman. Polylogarithmic independence fools AC^0 circuits. *Journal of the ACM*, 57(5), 2010.

[BV14] Eli Ben-Sasson and Emanuele Viola. Short PCPs with projection queries. In *Proc. 41st International Colloquium on Automata, Languages and Programming (ICALP)*, pages 163–173. 2014.

[BYRST02] Z. Bar-Yossef, O. Reingold, R. Shaltiel, and L. Trevisan. Streaming computation of combinatorial objects. In *Proc. 17th Annual IEEE Conference on Computational Complexity (CCC)*, pages 133–142, 2002.

[CKK+15] Ruiwen Chen, Valentine Kabanets, Antonina Kolokolova, Ronen Shaltiel, and David Zuckerman. Mining circuit lower bound proofs for meta-algorithms. *Computational Complexity*, 24(2):333–392, 2015.

[CL16] Kuan Cheng and Xin Li. Randomness extraction in AC0 and with small locality. *Electronic Colloquium on Computational Complexity: ECCC*, 23:18, 2016.

[CSS16] Ruiwen Chen, Rahul Santhanam, and Srikanth Srinivasan. Average-case lower bounds and satisfiability algorithms for small threshold circuits. In *Proc. 31st Annual IEEE Conference on Computational Complexity (CCC)*, pages 1:1–1:35, 2016.

[DGJ+10] Ilias Diakonikolas, Parikshit Gopalan, Ragesh Jaiswal, Rocco A. Servedio, and Emanuele Viola. Bounded independence fools halfspaces. *SIAM Journal of Computing*, 39(8):3441–3462, 2010.

[DP09] Devdatt Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009.

[GHR92] Mikael Goldmann, Johan Håstad, and Alexander Razborov. Majority gates vs. general weighted threshold gates. In *Proc. 7th Annual Structure in Complexity Theory Conference*, pages 2–13, 1992.

[GK98] Mikael Goldmann and Marek Karpinski. Simulating threshold circuits by majority circuits. *SIAM Journal of Computing*, 27(1):230–246, 1998.

[GKM15] Parikshit Gopalan, Daniel Kane, and Raghu Meka. Pseudorandomness via the discrete Fourier transform. In *Proc. 56th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 903–922. 2015.

[GMR13] Parikshit Gopalan, Raghu Meka, and Omer Reingold. Dnf sparsification and a faster deterministic counting algorithm. *Computational Complexity*, 22(2):275–310, 2013.

[Gol08] Oded Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, New York, NY, USA, 2008.

[GOWZ10] Parikshit Gopalan, Ryan O'Donnell, Yi Wu, and David Zuckerman. Fooling functions of halfspaces under product distributions. In *Proc. 25th Annual IEEE Conference on Computational Complexity (CCC)*, pages 223–234. 2010.

[GT91] Hans Dietmar Gröger and György" Turán. On linear decision trees computing boolean functions. In *Proc. 18th International Colloquium on Automata, Languages and Programming (ICALP)*, 1991.

[GVW15] Oded Goldreich, Emanuele Viola, and Avi Wigderson. On randomness extraction in AC0. In *Proc. 30th Annual IEEE Conference on Computational Complexity (CCC)*, pages 601–668, 2015.

[GW14] Oded Goldreich and Avi Widgerson. On derandomizing algorithms that err extremely rarely. In *Proc. 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 109–118. 2014. Full version available online at *Electronic Colloquium on Computational Complexity: ECCC*, 20:152 (Rev. 2), 2013.

[Hås94] Johan Håstad. On the size of weights for threshold gates. *SIAM Journal on Discrete Mathematics*, 7(3):484–492, 1994.

[Hea08] Alexander D. Healy. Randomness-efficient sampling within NC^1. *Computational Complexity*, 17(1):3–37, 2008.

[HKM12] Prahladh Harsha, Adam Klivans, and Raghu Meka. An invariance principle for polytopes. *Journal of the ACM*, 59(6):29:1–29:25, 2012.

[IKW02] Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: exponential time vs. probabilistic polynomial time. *Journal of Computer and System Sciences*, 65(4):672–694, 2002.

[IMP12] Russell Impagliazzo, William Matthews, and Ramamohan Paturi. A satisfiability algorithm for AC^0. In *Proc. 23rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 961–972, 2012.

[IMZ12] Russell Impagliazzo, Raghu Meka, and David Zuckerman. Pseudorandomness from shrinkage. In *Proc. 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 111–119. 2012.

[IPS97] Russell Impagliazzo, Ramamohan Paturi, and Michael E. Saks. Size-depth tradeoffs for threshold circuits. *SIAM Journal of Computing*, 26(3):693–707, 1997.

[IPS13] Russell Impagliazzo, Ramamohan Paturi, and Stefan Schneider. A satisfiability algorithm for sparse depth two threshold circuits. In *Proc. 54th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 479–488. 2013.

[IW98] R. Impagliazzo and A. Wigderson. Randomness vs. time: De-randomization under a uniform assumption. In *Proc. 39th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 734–, 1998.

[Kan11] Daniel M. Kane. A small PRG for polynomial threshold functions of Gaussians. In *Proc. 52nd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 257–266. 2011.

[Kan14] Daniel M. Kane. A pseudorandom generator for polynomial threshold functions of Gaussian with subpolynomial seed length. In *Proc. 29th Annual IEEE Conference on Computational Complexity (CCC)*, pages 217–228. 2014.

[KM15] Pravesh K. Kothari and Raghu Meka. Almost optimal pseudorandom generators for spherical caps. In *Proc. 47th Annual ACM Symposium on Theory of Computing (STOC)*, pages 247–256. 2015.

[KRS12] Zohar S. Karnin, Yuval Rabani, and Amir Shpilka. Explicit dimension reduction and its applications. *SIAM Journal of Computing*, 41(1):219–249, 2012.

[KW16] Daniel M. Kane and Ryan Williams. Super-linear gate and super-quadratic wire lower bounds for depth-two and depth-three threshold circuits. In *Proc. 48th Annual ACM Symposium on Theory of Computing (STOC)*, pages 633–643, 2016.

[LMN93] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, Fourier transform, and learnability. *Journal of the Association for Computing Machinery*, 40(3):607–620, 1993.

[MZ13] Raghu Meka and David Zuckerman. Pseudorandom generators for polynomial threshold functions. *SIAM Journal of Computing*, 42(3):1275–1301, 2013.

[Nis93] Noam Nisan. The communication complexity of threshold gates. In *Combinatorics, Paul Erdős is eighty, Vol. 1*, Bolyai Society Mathematical Studies, pages 301–315. 1993.

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

[NW94] Noam Nisan and Avi Wigderson. Hardness vs. randomness. *Journal of Computer and System Sciences*, 49(2):149–167, 1994.

[O'D14] Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

[PS94] Ramamohan Paturi and Michael E. Saks. Approximating threshold circuits by rational functions. *Information and Computation*, 112(2):257–272, 1994.

[ROS94] V. P. Roychowdhury, A. Orlitsky, and Kai-Yeung Siu. Lower bounds on threshold and related circuits via communication complexity. *IEEE Transactions on Information Theory*, 40(2):467–474, 1994.

[RRV02] Ran Raz, Omer Reingold, and Salil Vadhan. Extracting all the randomness and reducing the error in Trevisan's extractors. *Journal of Computer and System Sciences*, 65(1):97–128, 2002.

[RS10] Yuval Rabani and Amir Shpilka. Explicit construction of a small epsilon-net for linear threshold functions. *SIAM Journal of Computing*, 39(8):3501–3520, 2010.

[San10] Rahul Santhanam. Fighting perebor: new and improved algorithms for formula and QBF satisfiability. In *Proc. 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 183–192. 2010.

[Ser07] Rocco A. Servedio. Every linear threshold function has a low-weight approximator. *Computational Complexity*, 16(2):180–209, 2007.

[Smo90] Roman Smolensky. On interpolation by analytic functions with special properties and some weak lower bounds on the size of circuits with symmetric gates. In *Proc. 31st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 628–631, 1990.

[SSTT16] Takayuki Sakai, Kazuhisa Seto, Suguru Tamaki, and Junichi Teruyama. Bounded depth circuits with weighted symmetric gates: satisfiability, lower bounds and compression. In *Proc. 41st International Symposium on Mathematical Foundations of Computer Science*. 2016.

[ST12] K. Seto and S. Tamaki. A satisfiability algorithm and average-case hardness for formulas over the full binary basis. In *Proc. 27th Annual IEEE Conference on Computational Complexity (CCC)*, pages 107–116, 2012.

[ST17a] Rocco Servedio and Li-Yang Tan. Deterministic search for CNF satisfying assignments in almost polynomial time. In *Proc. 58th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2017.

[ST17b] Rocco Servedio and Li-Yang Tan. Learning and fooling depth-two threshold circuits. Unpublished manuscript, 2017.

[SW13] Rahul Santhanam and Ryan Williams. On medium-uniformity and circuit lower bounds. In *Proc. 28th Annual IEEE Conference on Computational Complexity (CCC)*, pages 15–23. 2013.

[Tam16] Suguru Tamaki. A satisfiability algorithm for depth two circuits with a sub-quadratic number of symmetric and threshold gates. *Electronic Colloquium on Computational Complexity: ECCC*, 23:100, 2016.

[Tel17] Roei Tell. Improved bounds for quantified derandomization of constant-depth circuits and polynomials. In *Proc. 32nd Annual IEEE Conference on Computational Complexity (CCC)*, pages 18:1 – 18:49, 2017.

[Tre01] Luca Trevisan. Extractors and pseudorandom generators. *Journal of the ACM*, 48(4):860–879, 2001.

[TS17] Amnon Ta-Shma. Explicit, almost optimal, epsilon-balanced codes. In *Proc. 49th Annual ACM Symposium on Theory of Computing (STOC)*, 2017.

[TX13] Luca Trevisan and TongKe Xue. A derandomized switching lemma and an improved derandomization of AC0. In *Proc. 28th Annual IEEE Conference on Computational Complexity (CCC)*, pages 242–247. 2013.

[Vad12] Salil P. Vadhan. *Pseudorandomness*. Foundations and Trends in Theoretical Computer Science. Now Publishers, 2012.

[Vio05] Emanuele Viola. The complexity of constructing pseudorandom generators from hard functions. *Computational Complexity*, 13(3-4):147–188, 2005.

[Wil11] Ryan Williams. Non-uniform ACC circuit lower bounds. In *Proc. 26th Annual IEEE Conference on Computational Complexity (CCC)*, pages 115–125. 2011.

[Wil13] Ryan Williams. Improving exhaustive search implies superpolynomial lower bounds. *SIAM Journal of Computing*, 42(3):1218–1244, 2013.

[Wil14a] Ryan Williams. Algorithms for circuits and circuits for algorithms: Connecting the tractable and intractable. In *Proc. International Congress of Mathematicians (ICM)*, pages 659–682, 2014.

[Wil14b] Ryan Williams. New algorithms and lower bounds for circuits with linear threshold gates. In *Proc. 55th Annual ACM Symposium on Theory of Computing (STOC)*, pages 194–202, 2014.

[Yao82] Andrew C. Yao. Theory and application of trapdoor functions. In *Proc. 23rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 80–91, 1982.

Appendix A Quantified derandomization and lower bounds

In this appendix we prove that "black-box" *quantified* derandomization of a class $\mathcal{C}$ yields lower bounds for $\mathcal{C}$, in the same way that standard derandomization does. For simplicity, we focus on the case of derandomization with one-sided error. Let us first recall the notion of a hitting-set generator, which yields a "black-box" quantified derandomization with one-sided error of a circuit class.

Definition A.1 *(hitting-set generator). Let $\mathcal{F} = \bigcup_{n\in\mathbb{N}} \mathcal{F}_n$, where for every $n \in \mathbb{N}$ it holds that $\mathcal{F}_n$ is a set of functions $\{0,1\}^n \to \{0,1\}$, and let $\ell : \mathbb{N} \to \mathbb{N}$. An algorithm H is a* hitting-set generator for $\mathcal{F}$ with seed length ℓ *if for every $n \in \mathbb{N}$ and every $f \in \mathcal{F}_n$ there exists $s \in \{0,1\}^{\ell(n)}$ such that $f(H(s)) = 1$.*

In the following proposition, we assume that there exists a hitting-set generator with non-trivial seed length $\ell(n) < n$ for circuits with $B(n) \geq 2^\ell$ exceptional inputs, and show that this implies lower bounds for the corresponding circuit class.

Proposition A.2 *(quantified derandomization implies lower bounds). Let $\ell : \mathbb{N} \to \mathbb{N}$ such that $\ell(n) < n$, and let $B : \mathbb{N} \to \mathbb{N}$ such that $B(n) \geq 2^{\ell(n)}$. Let $\mathcal{C}$ be a circuit class, and let $\mathcal{C}^{\leq B} \subseteq \mathcal{C}$ be the subclass of circuits that reject at most $B(n)$ of their inputs. Assume that there exists a $2^{O(\ell)}$-time computable hitting-set generator H with seed length ℓ for $\mathcal{C}^{\leq B}$. Then, there exists a function in $DTIME(2^{O(\ell(n))})$ that cannot be computed by* any *circuit in $\mathcal{C}$.*

Proof. The "hard" function for $\mathcal{C}$, denoted f, is the indicator function of $\{0,1\}^n \setminus \{H(s) : s \in \{0,1\}^{\ell(n)}\}$; that is, $f(x) = 0$ if and only if there exists $s \in \{0,1\}^{\ell(n)}$ such that $x = H(s)$. Note that any $C \in \mathcal{C}$ that computes f rejects at most $2^\ell \leq B(n)$ inputs, and thus $C \in \mathcal{C}^{\leq B}$. However, this means that H is a hitting-set generator for C, and so there exists $s \in \{0,1\}^{\ell(n)}$ such that $C(H(s)) = 1$. Since $f(H(s)) = 0$, we obtain a contradiction to the hypothesis that C computes f. ■

Appendix B Proof of a technical claim from Section 6

In the proof of Proposition 6.5, we omitted the proof of the following claim: For every $x \in \{0,1\}^n$ such that $x \neq 0^n$, the relative Hamming weight $\hat{x} = C(x)$ is at least $(1/3)^d$. The proof of this claim, which we now detail, follows from a standard property of tensor codes: If a code ECC has distance $\delta > 0$, then the tensor code of order d that is based on ECC has distance δ^d.

Claim B.1. *Let C be the circuit constructed in the proof of Proposition 6.5, and let $x \in \{0,1\}^n$ such that $x \neq 0^n$. Then, the relative Hamming weight $\hat{x} = C(x)$ is at least $(1/3)^d$.*

Proof. Recall that the code ECC maps any non-zero message of length m to a codeword of length $\bar{m}$ with at least $r \stackrel{\text{def}}{=} \bar{m}/3$ non-zero entries. Our hypothesis is that $x = M^{(0)}$ is not the all-zero message, and we will now prove that for each $i \in [d]$ it holds that $M^{(i)}$ has at least r^i non-zero entries. The proof is by induction, and will rely on a stronger induction hypothesis: We prove that for each $i \in \{0,...,d\}$ there exists $\vec{x}_{\geq i+1} \in [m]^{d-i}$ such that the number of vectors $\vec{x}_{\leq i} \in [\bar{m}]^i$ for which $M^{(i)}_{\vec{x}_{\leq i},\vec{x}_{\geq i+1}} \neq 0$ is at least r^i.

For the base case $i = 1$, note that by our hypothesis there exists $\vec{x} \in [m]^d$ such that $M^{(0)}_{\vec{x}} \neq 0$. Therefore, the m-bit vector $M^{(0)}_{\star,\vec{x}_{\geq 2}} = M^{(0)}_{1,\vec{x}_2,...,\vec{x}_d}, ..., M^{(0)}_{m,\vec{x}_2,...,\vec{x}_d}$ is non-zero. By the

properties of ECC it holds that $\texttt{ECC}\left(M^{(0)}_{\star,\vec{x}_{\geq 2}}\right)$ has at least r non-zero entries. The bits of $\texttt{ECC}\left(M^{(0)}_{\star,\vec{x}_{\geq 2}}\right)$ appear in $M^{(i)}$ in locations $(1,\vec{x}_2,...,\vec{x}_d),...,(\bar{m},\vec{x}_2,...,\vec{x}_d)$. Therefore, the claim is proved for $i=1$ with the vector $\vec{x}_{\geq 2}=\vec{x}_2,...,\vec{x}_d\in[m]^{d-1}$.

For the induction step, let $i\geq 2$. By the induction hypothesis, for some $\vec{x}_{\geq i}\in[m]^{d-(i-1)}$ there exist at least r^{i-1} vectors $\vec{x}^{(1)}_{\leq i-1},...,\vec{x}^{(r^{i-1})}_{\leq i-1}\in[\bar{m}]^{i-1}$ such that $M^{(i-1)}_{\vec{x}^{(j)}_{\leq i-1},\vec{x}_{\geq i}}\neq 0$ for all $j\in[r^{i-1}]$. Fix $j\in[r^{i-1}]$. Since $M^{(i-1)}_{\vec{x}^{(j)}_{\leq i-1},\vec{x}_{\geq i}}\neq 0$, it follows that the string $M^{(i-1)}_{\vec{x}^{(j)}_{\leq i-1},\star,\vec{x}^{(j)}_{\geq i+1}}=M^{(i-1)}_{\vec{x}^{(j)}_{\leq i-1},1,\vec{x}^{(j)}_{\geq i+1}},...,M^{(i-1)}_{\vec{x}^{(j)}_{\leq i-1},m,\vec{x}^{(j)}_{\geq i+1}}\in\{0,1\}^m$ is non-zero. Thus, by the properties of ECC, the string $\texttt{ECC}\left(M^{(i-1)}_{\vec{x}^{(j)}_{\leq i-1},\star,\vec{x}^{(j)}_{\geq i+1}}\right)$ contains at least r non-zero entries.

Now, for every $j\in[r^{i-1}]$, let $\vec{X}^{(j)}\stackrel{\text{def}}{=}\left\{\left(\vec{x}^{(j)}_{\leq i-1},1,\vec{x}^{(j)}_{\geq i+1}\right),...,\left(\vec{x}^{(j)}_{\leq i-1},\bar{m},\vec{x}^{(j)}_{\geq i+1}\right)\right\}$ be the set of $\bar{m}$ locations in $M^{(i)}$ in which the string $\texttt{ECC}\left(M^{(i-1)}_{\vec{x}^{(j)}_{\leq i-1},\star,\vec{x}^{(j)}_{\geq i+1}}\right)$ appears. Note that for every $j\neq j'\in[r^{i-1}]$ it holds that all locations in $X^{(j)}$ and $X^{(j')}$ are distinct; that is, for every $k,k'\in[\bar{m}]$ it holds that $\left(\vec{x}^{(j)}_{\leq i-1},k,\vec{x}^{(j)}_{\geq i+1}\right)\neq\left(\vec{x}^{(j')}_{\leq i-1},k',\vec{x}^{(j)}_{\geq i+1}\right)$. Since for each $j\in[r^{i-1}]$ it holds that $X^{(j)}$ contains at least r locations in which $M^{(i)}$ is non-zero, we deduce that $M^{(i)}$ has at least r^i non-zero entries. ■

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il