

Extractors using hardness amplification

Anindya De*

Luca Trevisan†

August 1, 2009

Abstract

Zimand [28] presented simple constructions of locally computable strong extractors whose analysis relies on the *direct product theorem* for one-way functions and on the *Blum-Micali-Yao* generator. For N -bit sources of entropy γN , his extractor has seed $O(\log^2 N)$ and extracts $N^{\gamma/3}$ random bits.

We show that his construction can be analyzed based solely on the direct product theorem for general functions. Using the direct product theorem of Impagliazzo et al. [7], we show that Zimand’s construction can extract $\tilde{\Omega}_\gamma(N^{1/3})$ random bits. (As in Zimand’s construction, the seed length is $O(\log^2 N)$ bits.)

We also show that a simplified construction can be analyzed based solely on the XOR lemma. Using Levin’s proof of the XOR lemma [9], we provide an alternative simpler construction of a locally computable extractor with seed length $O(\log^2 N)$ and output length $\tilde{\Omega}_\gamma(N^{1/3})$.

Finally, we show that the *derandomized direct product theorem* of Impagliazzo and Wigderson [8] can be used to derive a locally computable extractor construction with $O(\log N)$ seed length and $\tilde{\Omega}(N^{1/5})$ output length. Zimand describes a construction with $O(\log N)$ seed length and $O(2^{\sqrt{\log N}})$ output length.

Keywords: Extractors, Direct product theorems, Hardness amplification

*Computer Science Division, University of California, Berkeley, CA, USA. anindya@cs.berkeley.edu. Supported by “Berkeley Fellowship for Graduate Study”

†Computer Science Division, University of California, Berkeley, CA, USA. luca@cs.berkeley.edu. This material is based upon work supported by the National Science Foundation under grant No. CCF-0729137 and by the US-Israel BSF grant 2002246.

1 Introduction

Randomness extractors, defined by Nisan and Zuckerman [29, 14] are a fundamental primitive with several applications in pseudorandomness and derandomization. A function $Ext : \{0, 1\}^N \times \{0, 1\}^t \rightarrow \{0, 1\}^m$ is a (K, ϵ) -extractor if, for every random variable X of min-entropy at least K , the distribution $Ext(X, U_t)$ has statistical distance at most ϵ from the uniform distribution over $\{0, 1\}^m$.¹

Besides their original applications to extract randomness from weak random sources and as primitives inside pseudorandom generators for space bounded computations, extractors have found several other applications. As surveyed in [13, 18] extractors are related to hashing and error-correcting codes, and have applications to pseudorandomness and hardness of approximation.

Extractors have also found several applications in cryptography, for example in unconditionally secure cryptographic constructions in the bounded-storage model [11, 1, 10]. For such application, it is particularly desirable to have *locally computable* extractors, in which a bit of the output can be computed by only looking at the seed and at *poly* $\log n$ bits of the input. (The weaker notion of *online* extractors, however, is sufficient.)

The starting point of our paper is Zimand’s [28] simple construction of a locally computable extractor based on the Blum-Micali-Yao pseudorandom generator, and his analysis via the reconstruction approach of [23].

The idea of the reconstruction approach to the analysis of extractors is the following. Suppose we want to prove that $Ext : \{0, 1\}^N \times \{0, 1\}^t \rightarrow \{0, 1\}^m$ is a (K, ϵ) extractor. Then, towards a contradiction, we suppose there is a test $T : \{0, 1\}^m \rightarrow \{0, 1\}$ and a random variable X of min entropy at least K such that

$$|\mathbb{P}[T(Ext(X, U_t)) = 1] - \mathbb{P}[T(U_m) = 1]| > \epsilon$$

In particular, there is a probability at least $\epsilon/2$ when sampling from X of selecting a bad x such that

$$|\mathbb{P}[T(Ext(x, U_t)) = 1] - \mathbb{P}[T(U_m) = 1]| > \frac{\epsilon}{2}$$

At this point, one uses properties of the construction to show that if x is bad as above, x can be *reconstructed* given T and a r -bit string of “advice.” This means that there can be at most 2^r bad strings x , and if X has min-entropy K then the probability of sampling a bad x is at most $2^r/2^K$, which is a contradiction if $2^K > 2^{r+1}/\epsilon$.

In Zimand’s extractor construction, one thinks of a sample from X as specifying a cyclic permutation $p : \{0, 1\}^n \rightarrow \{0, 1\}^n$ (where n is roughly $\log N$), then lets \bar{p} be a permutation obtained from p via a *hardness amplification* procedure, so that the ability to invert \bar{p} on a small α fraction of inputs implies the ability of invert p on a large $1 - \delta$ fraction of inputs. Then the output of the extractor, for seed z , is $BM(Y(\bar{p}, z))$, the Blum-Micali-Yao generator applied to permutation \bar{p} with seed z . If

¹We use U_n to denote the uniform distribution over $\{0, 1\}^n$, and recall that a distribution X is said to have min-entropy at least K if for every a we have $\mathbb{P}[X = a] \leq 2^{-K}$. Two random variables Y, Z ranging over the same universe $\{0, 1\}^m$ have distance at most ϵ in statistical distance if for every statistical test $T : \{0, 1\}^m \rightarrow \{0, 1\}$ we have

$$|\mathbb{P}[T(Y) = 1] - \mathbb{P}[T(Z) = 1]| \leq \epsilon$$

a test T distinguishes the output of the extractor from the uniform distribution, then there is an algorithm that, using T , can invert \bar{p} on a noticeable fraction of inputs, and hence p on nearly all inputs. The proof is completed by presenting a counting argument showing an upper bound on the number of permutations that can be easily inverted on nearly all inputs.

Zimand's extractor uses a seed of length $O(\log^2 N)$ and, for a source of entropy γN , the output length is $N^{\gamma/3}$ bits.

We show that, by using only direct product theorems and XOR lemmas, we can improve the output length to roughly $N^{1/3}$. This is true both for Zimand's original construction², as well as for a streamlined version we describe below. The streamlined version is essentially the same construction as the locally computable extractor of Dziembowski and Maurer [3]. Our analysis via Levin's XOR lemma is rather different from the one in [3] which is based on information-theoretic arguments.

Using the *derandomized* direct product theorem of Impagliazzo and Wigderson [8], we give a construction in which the seed length reduces to $O(\log N)$, but the output length reduces to $N^{1/5}$.

Our Constructions

Consider the following approach. View the sample from the weak random source as a boolean function $f : [N] \rightarrow \{0, 1\}$, and suppose that the extractor simply outputs the sequence

$$f(x), f(x+1), \dots, f(x+m-1)$$

where $x \in [N]$ is determined by the seed, and sums are computed mod N . Then, by standard arguments, if T is a test that distinguishes the output of the extractor from the uniform distribution with distinguishing probability ϵ , then there is a predictor P , derived from T , and $i \leq m$ such that

$$\mathbb{P}[P(x, f(x-1), \dots, f(x-i)) = f(x)] \geq \frac{1}{2} + \frac{\epsilon}{m} \tag{1}$$

Note that if the right-hand side of (1) were $1 - \delta$ for some small δ , instead of $1/2 + \epsilon/m$, then we could easily deduce that f can be described using about $m + \delta N + H(\delta) \cdot N$ bits (where $H()$ is the entropy function), and so we would be done.

To complete the argument, given the function $f : [N] \rightarrow \{0, 1\}$ that we sample from the random source, we define the function $\bar{f} : [N]^k \rightarrow \{0, 1\}$ as

$$\bar{f}(x_1, \dots, x_k) := \bigoplus_{i=1}^k f(x_i)$$

where $k \approx \log N$, and our extractor outputs

$$\bar{f}(\bar{x}), \bar{f}(\bar{x} + \mathbf{1}), \dots, \bar{f}(\bar{x} + \mathbf{m} - \mathbf{1})$$

where $\bar{x} = (x_1, \dots, x_k) \in [N]^k$ is selected by the seed of the extractor, \mathbf{j} is the vector (j, \dots, j) , and sums are coordinate-wise, and mod N .

²We actually do not show an improved analysis for this specific construction by Zimand but rather for the second construction in the same paper which achieves exactly the same parameters. Our improved analysis works equally for both the constructions but is slightly notationally cumbersome for the first one

If T is a test that has distinguishing probability ϵ for our extractor, then there is a predictor P based on T such that

$$\mathbb{P}[P(\bar{x}, \bar{f}(\bar{x} - \mathbf{1}), \dots, \bar{f}(\bar{x} - \mathbf{i})) = \bar{f}(\bar{x})] \geq \frac{1}{2} + \frac{\epsilon}{m} \quad (2)$$

from which we can use the proof of the XOR lemma to argue that, using P and some advice, we can construct a predictor P' such that

$$\mathbb{P}[P'(x, f(x - 1), \dots, f(x - i)) = f(x)] \geq 1 - \delta \quad (3)$$

and now we are done. Notice that we cannot use standard XOR lemmas as a black box in order to go from (2) to (3), because the standard theory deals with a predictor that is only given x , rather than $x, f(x - 1), \dots, f(x - i)$. The proofs, however, can easily be modified at the cost of extra non-uniformity. To adapt, for example, Levin's proof of the XOR Lemma, we see that, in order to predict $f(x)$, it is enough to evaluate P at $O(m^2/\epsilon^2)$ points \bar{x} , each of them containing x in a certain coordinate and fixed values everywhere else. For each such point, $F(\bar{x} - \mathbf{1}), \dots, F(\bar{x} - \mathbf{i})$ can be specified using $i \cdot (k - 1) \leq mk$ bits of advice. Overall, we need m^3k/ϵ^2 bits of advice, which is why we can only afford the output length m to be the cubed root of the entropy. The seed length is $k \log N$, which is $O(\log^2 N)$.

This type of analysis is robust to various changes to the construction. For example, we can view a sample from the weak random source as a function $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$, define

$$\bar{f}(x_1, \dots, x_k) := f(x_1), \dots, f(x_k) ,$$

View the seed as specifying an input \bar{x} for $\bar{f}()$ and a boolean vector r of the same length, and define the output of the extractor as

$$\langle \bar{f}(\bar{x}), r \rangle, \langle \bar{f}(\bar{x} + \mathbf{1}), r \rangle, \dots, \langle \bar{f}(\bar{x} + \mathbf{m} - \mathbf{1}), r \rangle \quad (4)$$

Then using appropriate versions of Goldreich-Levin and of the direct product lemma of Impagliazzo et al. [7], we can show that the construction is an extractor provided that m is about $N^{1/3}$ ³. Construction (4) is precisely the second construction by Zimand [28].

By applying the *derandomized* direct product theorem of Impagliazzo and Wigderson [8], we are able to reduce the seed length to $O(\log N)$, but our reconstruction step requires more non-uniformity, and so the output length of the resulting construction is only about $N^{1/5}$.

Organization of the paper

In section 2, we present some notations which shall be used throughout the paper and an overview of the techniques recurrent in the proofs of all the three constructions. Section 3 presents the first of our constructions. Its proof of correctness is self contained. Appendix A describes the construction by Zimand [28] and presents an improved analysis of the same. In Appendix B, we present a new extractor which can be seen as a derandomized version of the first two extractors.

³Even using the 'concatenation lemma' of Goldreich et al. [5] which is a much more non-uniform version of the direct product theorem, we get $m = N^{1/10}$ for which is better than Zimand's analysis for entropy rates < 0.3

2 Preliminaries and overview of proofs

Notations and definitions

The following notations are used throughout the paper. A tuple (y_1, y_2, \dots, y_k) is denoted by $\otimes_{i=1}^k y_i$. The concatenation of two strings x and y is denoted by $x \circ y$. If x and y are tuples, then $x \circ y$ represents the bigger tuple formed by concatenating x and y . The uniform distribution on $\{0, 1\}^n$ is denoted by U_n . For $z_1, \dots, z_k \in \{0, 1\}$, $\oplus_{i=1}^k z_i$ denotes the XOR of z_1, \dots, z_k . Statistical distance between two distributions D_1 and D_2 is denoted by $\|D_1 - D_2\|$

Next, we define extractors as well as a stronger variant called strong extractors.

Definition 2.1 [16, 29] *Ext* : $\{0, 1\}^N \times \{0, 1\}^t \rightarrow \{0, 1\}^m$ is said to be a (K, ϵ) extractor if for every random variable X with min-entropy at least K , $\|Ext(X, U_t) - U_m\| \leq \epsilon$. *Ext* is said to be a strong extractor if $\|U_t \circ Ext(X, U_t) - U_{t+m}\| \leq \epsilon$. Here both the U_t refer to the same sampling of the uniform distribution.

In the above definition, t is referred to as seed length, m as the output length and ϵ as the error of the extractor.

General paradigm of construction

All the three extractors can be described in the following general model. Let $Ext : \{0, 1\}^N \times \{0, 1\}^t \rightarrow \{0, 1\}^m$ be the extractor (terminology is the same as Definition 2.1) with X representing the weak random source and \bar{y} the seed. X is treated as truth table of a function $X : \{0, 1\}^n \rightarrow \{0, 1\}^l$ ($l = 1$ in the first and the third constructions and $l = n$ in the second construction). This implies that n is logarithmic in the input length N and more precisely $N = l2^n$. Further, we associate a cyclic group of size 2^n with $\{0, 1\}^n$ (This can be any ordering of the elements in $\{0, 1\}^n$ except that the addition in the group should be efficiently computable). To make it easier to remind us that X is treated as truth table of a function, the corresponding function shall henceforth be called f . The seed \bar{y} is divided into two chunks i.e. $\bar{y} = \bar{x} \circ \bar{z}$. \bar{x} is called the input chunk and \bar{z} is called the encoding chunk. Also, let k is a parameter of the construction such that $|\bar{x}| = g(n, k)$ and $|\bar{z}| = h(n, k)$ and hence $t = g(n, k) + h(n, k)$. Ext is specified by two functions namely $Exp : \{0, 1\}^{g(n, k)} \rightarrow (\{0, 1\}^n)^k$ and $Com : (\{0, 1\}^l)^k \times \{0, 1\}^{h(n, k)} \rightarrow \{0, 1\}$. Ext computes the output as follows

- On input $(X, \bar{y}) \equiv (f, \bar{x} \circ \bar{z})$, Ext first computes $Exp(\bar{x}) = (x_1, x_2, x_3, \dots, x_k)$ which gives k candidate inputs for the function f .
- Subsequently, the i^{th} bit of the output is computed by combining the evaluation of f at shifts of (x_1, \dots, x_k) using Com . More precisely, the i^{th} bit is given by $Com(\otimes_{j=1}^k f(x_j + i - 1), \bar{z})$.

Our constructions differ from each other in the definition of the functions Exp and Com . It can be easily seen that as long as Exp and Com are efficiently computable i.e. both of them are computable in $poly(n, k)$ time and $k = O(n)$, the extractors shall be locally computable. This is true for all our constructions.

Proofs in the reconstruction paradigm

We now show the steps (following the reconstruction paradigm) which are used in the proof of correctness of all the constructions. We first note that proving $Ext : \{0, 1\}^N \times \{0, 1\}^t \rightarrow \{0, 1\}^m$ is

a $(\gamma N, 2\epsilon)$ strong extractor is equivalent to proving that for every boolean function $T : \{0, 1\}^{m+t} \rightarrow \{0, 1\}$ and random variable X of min-entropy at least γN

$$|Pr_{f \in X, \bar{y} \in U_i}[T(y, Ext(f, \bar{y})) = 1] - Pr_{u \in U_{t+m}}[T(u) = 1]| \leq 2\epsilon \quad (5)$$

We had earlier noted the following fact which we formally state below.

Observation 2.2 *In order to prove equation (5), it suffices to prove that for any $T : \{0, 1\}^{m+t} \rightarrow \{0, 1\}$, there are at most $\epsilon 2^{\gamma N}$ functions f such that*

$$|Pr_{\bar{y} \in U_i}[T(y, Ext(f, \bar{y})) = 1] - Pr_{u \in U_{t+m}}[T(u) = 1]| > \epsilon \quad (6)$$

In order to bound the number of functions which satisfy (6), we use the reconstruction approach in [23]⁴ (and more generally used in the context of pseudorandom generators in [2, 15]). In particular, we show that given any f which satisfies (6), we can get a circuit C_f (not necessarily small) which predicts value of f by querying f at some related points. More precisely, we show that for some $m > i \geq 0$, using c bits of advice, we can construct C_f which satisfies (7) for some $s \leq \frac{1}{2}$.

$$Pr_{x \in U_n}[C_f(x, \otimes_{j=1}^i f(x-j)) = f(x)] \geq 1 - s \quad (7)$$

The next lemma shows how such a circuit C_f can be used to bound the number of functions f satisfying (6).

Lemma 2.3 *If for every f satisfying (6), using c bits of advice, we can get a circuit C_f satisfying (7) for some $s \leq \frac{1}{2}$, then there are at most $2^{c+2^n(sl+H(s))+ml}$ functions satisfying (6).*

Proof: Let the set BAD consist of points $x \in \{0, 1\}^n$ such that $C_f(x, \otimes_{j=1}^i f(x-j)) \neq f(x)$. Since the size of the set BAD is at most $s2^n$, to fully specify the set, we require at most $\log_2 S$ bits where $S = \sum_{i=0}^{s2^n} \binom{2^n}{i}$. Further, to specify the value of f on the set BAD , we require at most $sl2^n$ bits. We now note that if we are given the value of f on any consecutive i points (say $[0, \dots, i-1]$), which requires at most il bits, then using the circuit C_f , the set BAD and the value of f on points in BAD , one can fully specify f . We also use the following standard fact. (Log is taken base 2 unless mentioned otherwise)

Fact 2.4 *For $s \leq \frac{1}{2}$, $\sum_{i=0}^{s2^n} \binom{2^n}{i} \leq 2^{H(s)2^n}$ where $H(s) = -s \log s - (1-s) \log(1-s)$.*

Hence, we see that if we are given that f satisfies (6), then using T and $c + 2^n(s + H(s)) + il$ bits of advice, we can exactly specify f . Hence for any particular T , (using $i < m$) we get that there are at most $2^{c+2^n(sl+H(s))+ml}$ functions satisfying (6). ■

In light of lemma 2.3, given f satisfying (6), we should use T to construct a circuit C_f satisfying (7) with as minimum advice and as small s as possible. We first use the standard hybrid argument and Yao's distinguisher versus predictor argument to get a circuit which is a 'next-element' predictor. In particular, we create a circuit which predicts a particular position in the output of the extractor with some advantage over a random guess when given as input the value of the random seed as well as all the bits in the output preceding the bit to be predicted. The argument is by now standard and can be found in several places including [23, 22, 19]. We do not redo the argument here but simply state the final result.

⁴This particular instance of reconstruction paradigm was used in context of extractors by Zimand [28] and earlier in context of pseudorandom generators by Blum, Micali and Yao [2, 27].

Lemma 2.5 *Let f be any function satisfying (6) and $Ext(f, \bar{y})_i$ be the i^{th} bit of the output. Then using $m + \log m + 3$ bits of advice, we can get a circuit T_2 such that for some $0 \leq i < m$, f satisfies (8).*

$$Pr_{\bar{y} \in U_i} [T_2(\bar{y}, \otimes_{j=1}^{m-i-1} Ext(f, \bar{y})_j) = Ext(f, \bar{y})_{m-i}] > \frac{1}{2} + \frac{\epsilon}{m} \quad (8)$$

The proof of correctness of all our constructions start from the above equation and use more advice to finally get a circuit C_f satisfying (7). We now describe the individual constructions and their respective proofs of correctness.

3 Extractor from XOR lemma

Description of the construction

$Ext : \{0, 1\}^{2^n} \times \{0, 1\}^{kn} \rightarrow \{0, 1\}^m$ is defined as follows. On input (f, \bar{y}) , the seed \bar{y} is partitioned into k chunks of length n - call it $(x_1, x_2, x_3, \dots, x_k)$. The source f is treated as truth table of a function from $\{0, 1\}^n$ to $\{0, 1\}$. Then the i^{th} bit of the output is given by the bitwise XOR of $f(x_1 + i - 1), \dots, f(x_k + i - 1)$ i.e. $Ext(f, \bar{y})_i = \oplus_{j=1}^k f(x_j + i - 1)$. In terminology of the last section, $N = 2^n$, $g(k, n) = kn$ and $h(k, n) = 0$. Note that there is no encoding chunk in the seed and the entire seed is the input chunk. Further, the function Exp simply partitions a string of length kn into k chunks of length n while the function Com computes a bitwise XOR of its first input (the second input is the empty string).

Difference from construction in [3]

As we have mentioned before, the construction in [3] is very similar though we have some minor simplifications. The extractor in [3] $Ext' : (\{0, 1\}^{N+m-1})^k \times \{0, 1\}^{k \log N} \rightarrow \{0, 1\}^m$ can be described as follows. The weak source is treated as truth table of k functions f_1, \dots, f_k such that for each $j \in [k]$, $f_j : [N + m - 1] \rightarrow \{0, 1\}$. The seed is divided into k chunks l_1, \dots, l_k such that each l_j can be treated as an element in $[N]$. The i^{th} bit of the output is computed as $\oplus_{j=1}^k f_j(l_j + i - 1)$. Thus, we avoid a minor complication of not having to divide the source into chunks and the . Our proof can be modified to work in this case as well at the cost of making it more cumbersome while conceptually remaining the same. However, the main difference is that we come up with an entirely different proof from the one in [3].

Main theorem and Proof of correctness

Theorem 3.1 *The function $Ext : \{0, 1\}^{2^n} \times \{0, 1\}^{kn} \rightarrow \{0, 1\}^m$ is a $(\gamma 2^n, 2\epsilon)$ strong extractor for a (constant) $\gamma > 0$, $\epsilon \geq 2^{-\frac{n}{7}}$, $m = \frac{\epsilon^{\frac{2}{3}} 2^{\frac{n}{3}}}{n^2}$ and seed length $kn = O\left(\frac{n \log \frac{m}{\epsilon}}{\gamma^2}\right)$*

Before proving Theorem 3.1, we see an immediate corollary of the above theorem with parameters of interest.

Corollary 3.2 *The function Ext as defined above is a $(\gamma 2^n, 2\epsilon)$ strong extractor for a (constant) $\gamma > 0$, $2\epsilon = 2^{-n^{\frac{1}{4}}}$, $m = 2^{\frac{n}{3} - \sqrt{n}}$ and seed length $kn = O\left(\frac{n^2}{\gamma^2}\right)$.*

In order to prove Theorem 3.1, we first state the following main technical lemma of this section and then see how Theorem 3.1 follows from it. Subsequently, we prove the lemma.

Lemma 3.3 *Let $T : \{0, 1\}^{m+kn} \rightarrow \{0, 1\}$ and $f : \{0, 1\}^n \rightarrow \{0, 1\}$ such that (6) holds. Also, let $1 > \delta > 0$ be such that $\delta^k \leq \frac{\epsilon}{m}$ and $m \geq nk$. Then with at most $\frac{6nk^2m^3}{\epsilon^2}$ bits of advice, we can get a circuit C_f such that $\Pr_{x_1 \in U_n}[C_f(x_1, \otimes_{j=1}^i f(x_1 - j)) = f(x_1)] \geq \frac{1+\delta}{2}$*

Before we formally prove Theorem 3.1 using Lemma 3.3, it is useful to mention that just for this section, an application of δ is meaningful when it is close to 1 rather than 0. As can be seen from Lemma 3.3, we construct a circuit C_f which has correlation δ with f and hence we would like $1 - \delta$ to be small. This is different from Section 1 and subsequent sections where we want to construct a circuit C_f which computes f with probability $1 - \delta$ and hence we would like δ to be close to 0.

Proof: [of Theorem 3.1] In light of Observation 2.2, we note that it is sufficient to prove that for any statistical test $T : \{0, 1\}^{m+kn} \rightarrow \{0, 1\}$, the number of functions f satisfying (6) is at most $\epsilon 2^{\gamma N}$. Let δ be such that $\frac{1-\delta}{2} = \min\{10^{-3}, \frac{\gamma^2}{4}\}$. Also putting $k = \frac{C \log \frac{m}{\epsilon}}{\gamma^2} = O\left(\frac{n}{\gamma^2}\right)$ for some appropriate constant C clearly satisfies $\delta^k \leq \frac{\epsilon}{m}$. Further, $m = 2^{\Omega(n)}$ while $nk = O\left(\frac{n^2}{\gamma^2}\right)$. So, clearly $m \geq nk$ for constant γ and sufficiently large n . With this, we satisfy the conditions for applying lemma 3.3 and hence with $\frac{6nk^2m^3}{\epsilon^2}$ bits of advice, we can get a circuit C_f satisfying (7) with $s = \frac{1-\delta}{2}$. Using lemma 2.3, we can say that for any test T , the total number of functions satisfying (6) is at most $2^{\frac{6nk^2m^3}{\epsilon^2} + (\frac{1-\delta}{2} + H(\frac{1-\delta}{2}))2^n + m}$. We now use the following fact

Fact 3.4 *For any $0 \leq \alpha \leq 10^{-3}$, $\alpha + H(\alpha) \leq \sqrt{\alpha}$*

Putting everything together now, we get that the total number of functions satisfying (6) is at most (we consider the case when $\gamma > 0$ is a constant and n is large enough integer).

$$2^{\frac{6nk^2m^3}{\epsilon^2} + (\frac{1-\delta}{2} + H(\frac{1-\delta}{2}))2^n + m} \leq 2^{O(\frac{2^n}{n^3\gamma^4})} 2^{\frac{\gamma}{2}2^n} 2^{2^{\frac{n}{3}}} \leq 2^{-\frac{n}{7}} 2^{\gamma 2^n} \leq \epsilon 2^{\gamma 2^n}$$

■

Proof: [of Lemma 3.3] Using lemma 2.5, we get that for any f such that (6) holds, using $m + \log m + 3$ bits of advice, we can get a circuit T_2 such that

$$\Pr_{x_1, \dots, x_k}[T_2(x_1, \dots, x_k, \oplus_{j=1}^k f(x_j), \dots, \oplus_{j=1}^k f(x_j + m - i - 2)) = \oplus_{j=1}^k f(x_j + m - i - 1)] > \frac{1}{2} + \frac{\epsilon}{m}$$

In the above, x_1, x_2, \dots, x_k are independent random variables drawn from U_n . Unless otherwise stated, in this section, any variable picked randomly is picked from the uniform distribution (The domain shall be evident from the context). We now introduce some changes in the notation so as to make it more convenient. First of all, we note that $m - i - 1$ can be replaced by i as i runs from 0 to $m - 1$. Further, we can assume that the first k arguments in the input are changed from x_j to $x_j + i$ for all $1 \leq j \leq k$ and hence we get a circuit C such that

$$\Pr_{x_1, \dots, x_k}[C(x_1, \dots, x_k, \oplus_{j=1}^k f(x_j - i), \dots, \oplus_{j=1}^k f(x_j - 1)) = \oplus_{j=1}^k f(x_j)] > \frac{1}{2} + \frac{\epsilon}{m}$$

In this proof, we closely follow the proof of XOR lemma due to Levin [9] as presented in [5]. As is done there, for convenience, we change the range of f from $\{0, 1\}$ to $\{-1, 1\}$ i.e. $f(x)$ now changes

to $(-1)^{f(x)}$. With this notational change, parity changes to product and prediction changes to correlation i.e.

$$\mathbb{E}_{x_1, \dots, x_k} \left[\prod_{j=1}^k f(x_j) C(x_1, \dots, x_k, \prod_{j=1}^k f(x_j - i), \dots, \prod_{j=1}^k f(x_j - 1)) \right] > \frac{2\epsilon}{m}$$

In order to simplify the notation further, we make one more change. For any tuple $(x_1, x_2, \dots, x_t) = \bar{x}$, $\prod_{j=1}^t f(x_j - s)$ is denoted by $\bar{f}(\bar{x} - s)$. Let $\bar{x} = (x_1, x_2, \dots, x_k)$ and using the notation introduced earlier for denoting tuples, we get

$$\mathbb{E}_{\bar{x}} [\bar{f}(\bar{x}) C(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j))] > \frac{2\epsilon}{m}$$

Let δ and η be such that $\delta^k \leq \frac{\epsilon}{m}$ and $\eta = \frac{\epsilon}{km}$. Then the above equation can be rewritten as

$$\mathbb{E}_{\bar{x}} [\bar{f}(\bar{x}) C(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j))] > \delta^k + k\eta \quad (9)$$

Further, we can write \bar{x} as $x_1 \circ \bar{y}$ where $x_1 \in \{0, 1\}^n$ and $\bar{y} \in (\{0, 1\}^n)^{k-1}$ and then the above can be rewritten as

$$\mathbb{E}_{x_1 \in U_n} [f(x_1) \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))] > \delta^k + k\eta \quad (10)$$

where $\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j)) = \mathbb{E}_{\bar{y} \in U_{(k-1)n}} [\bar{f}(\bar{y}) C(x_1 \circ \bar{y}, \otimes_{j=1}^i f(x_1 - j) \bar{f}(\bar{y} - j))]$. At this stage, there are the following two possibilities.

1. $\forall x_1, \left| \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j)) \right| \leq \delta^{k-1} + (k-1)\eta.$
2. $\exists x_1$ such that $\left| \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j)) \right| > \delta^{k-1} + (k-1)\eta.$

The following lemma shows how to construct the circuit in (7) in the first case. The second case follows by an inductive argument.

Lemma 3.5 *If for all x_1 , $\left| \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j)) \right| \leq \delta^{k-1} + (k-1)\eta$, then with $\frac{4nm}{\eta^2} + \log\left(\frac{4n}{\eta^2}\right) + 1$ bits of advice, we can get a circuit $C_f : \{0, 1\}^n \times \{0, 1\}^i \rightarrow \{-1, 1\}$ such that*

$$\mathbb{E}_{x_1} [f(x_1) C_f(x_1, \otimes_{j=1}^i f(x_1 - j))] > \delta \quad (11)$$

Proof: Let $\Gamma_1(x_1, \otimes_{j=1}^i f(x_1 - j)) = \frac{\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))}{\delta^{k-1} + (k-1)\eta} \in [-1, 1]$. We note that (10) says that $\Gamma_1(x_1, \otimes_{j=1}^i f(x_1 - j))$ has high correlation with $f(x_1)$ and hence if we could compute Γ_1 , then we could compute $f(x_1)$ with high probability. Since computing Γ_1 looks unlikely (without using 2^n bits of advice), we will approximate Γ_1 and still manage to compute f with high probability. In particular, we define a circuit C_1 such that for every x_1 , C_1 approximates $\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))$ within an additive error of η when given input x_1 and $\otimes_{j=1}^i f(x_1 - j)$. To do this, C_1 picks up $q = \frac{2n}{\eta^2}$ elements independently at random from $(\{0, 1\}^n)^{(k-1)}$. Call these elements $\bar{w}_1, \dots, \bar{w}_q$. C_1 then takes $\otimes_{j=0}^i \bar{f}(\bar{w}_l - j)$ for $l \in [q]$ as advice. Subsequently, it computes the function Γ_2 which is defined as follows (Note that Γ_2 depends upon \bar{w}_i 's and the corresponding advice though \bar{w}_i 's are not explicitly included in the argument).

$$\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j)) = \mathbb{E}_{l \in [q]} [\bar{f}(\bar{w}_l) C(x_1 \circ \bar{w}_l, \otimes_{j=1}^i f(x_1 - j) \bar{f}(\bar{w}_l - j))]$$

By Chernoff bound, we can say the following is true for all x_1 (the probability is over the random choices of \bar{w}_l for $l \in [q]$).

$$Pr[|\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j)) - \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))| > \eta] < 2^{-n}$$

We would like our estimate of $\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))$ to have absolute value bounded by $\delta^{k-1} + (k-1)\eta$. Hence, we define Γ_3 as follows.

1. If $|\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j))| \leq \delta^{k-1} + (k-1)\eta$, $\Gamma_3(x_1, \otimes_{j=1}^i f(x_1 - j)) = \Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j))$
2. If not, $\Gamma_3(x_1, \otimes_{j=1}^i f(x_1 - j)) = \frac{|\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j))|}{|\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j))|} (\delta^{k-1} + (k-1)\eta)$

The final output of $C_1(x_1, \otimes_{j=1}^i f(x_1 - j))$ is $\Gamma_3(x_1, \otimes_{j=1}^i f(x_1 - j))$. Since Γ_3 is definitely at least as good a approximation of Γ as Γ_2 is, we can say the following (the probability is again over the random choices of \bar{w}_l for $l \in [q]$ and as before \bar{w}_l is not explicitly included in the argument).

$$Pr[|\Gamma_3(x_1, \otimes_{j=1}^i f(x_1 - j)) - \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))| > \eta] < 2^{-n}$$

By a simple union bound, we can see that there exists a q -tuple $\otimes_{l=1}^q \bar{w}_l$ is such that for all x_1 , $|\Gamma_3(x_1, \otimes_{j=1}^i f(x_1 - j)) - \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))| \leq \eta$. Hence with $qn(k-1) \leq \frac{2n^2k}{\eta^2}$ bits of advice, we can get such a tuple $\otimes_{l=1}^q \bar{w}_l$. Further, the advice required for getting $\otimes_{j=0}^i \bar{f}(\bar{w}_l - j)$ for each $l \in [q]$ is $(i+1)q \leq \frac{2nm}{\eta^2}$ bits. So, we hardwire these ‘good’ values of \bar{w}_l and $\otimes_{j=0}^i \bar{f}(\bar{w}_l - j)$ into C_1 (i.e. instead of taking random choices, it now works with these hardwired values) and we can say that

$$\mathbb{E}_{x_1 \in U_n}[f(x_1)C_1(x_1, \otimes_{j=1}^i f(x_1 - j))] \geq \mathbb{E}_{x_1 \in U_n}[f(x_1)\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))] - \eta > \delta^k + (k-1)\eta \quad (12)$$

The above fact uses that the range of f is $[-1, 1]$. We now define $C_2(x_1, \otimes_{j=1}^i f(x_1 - j)) = \frac{C_1(x_1, \otimes_{j=1}^i f(x_1 - j))}{\delta^{k-1} + (k-1)\eta}$. Note that the output of C_2 is in $[-1, 1]$ and hence by (12), we can say (using $\delta \leq 1$)

$$\mathbb{E}_{x_1}[f(x_1)C_2(x_1, \otimes_{j=1}^i f(x_1 - j))] > \frac{\delta^k + (k-1)\eta}{\delta^{k-1} + (k-1)\eta} \geq \delta \quad (13)$$

C_2 is almost the circuit C_f we require except its output is in $[-1, 1]$ rather than $\{-1, 1\}$. To rectify this, we define a randomized circuit C_3 which computes $r = C_2(x_1, \otimes_{j=1}^i f(x_1 - j))$ and then outputs 1 with probability $\frac{1+r}{2}$ and -1 with probability $\frac{1-r}{2}$ otherwise. Clearly this randomized circuit C_3 has the same correlation with $f(x_1)$ as C_2 does. To fix the randomness of the circuit C_3 and to get C_f , we observe that the output of C_2 can only be in multiples of $\frac{\eta^2}{2n(\delta^{k-1} + (k-1)\eta)}$. Since the output is in the interval $[-1, 1]$, it suffices to pick a random string $\lceil \log \frac{4n(\delta^{k-1} + (k-1)\eta)}{\eta^2} \rceil$ bits long (rather than a random number in $[-1, 1]$). Hence by fixing this randomness using $\lceil \log \frac{4n}{\eta^2} \rceil \leq \log \frac{4n}{\eta^2} + 1$ bits of advice, we get a circuit C_f which satisfies (11)⁵. Clearly, the total amount of advice required is at most $\frac{2n(m+nk)}{\eta^2} + \log \left(\frac{4n}{\eta^2} \right) + 1$ bits. Using $m \geq nk$, we get the bound on the advice stated in the lemma. ■

⁵We remove the factor $\log(\delta^{k-1} + (k-1)\eta)$ in calculating the advice because $(\delta^{k-1} + (k-1)\eta)$ is at most 1 and hence what we are calculating is an upper bound on the advice

Hence, in the first case, we get a circuit C_f such that its expected correlation with f is greater than δ . Changing the $\{-1, 1\}$ notation to $\{0, 1\}$ notation, we get that

$$Pr_{x_1 \in U_n}[C_f(x_1, \otimes_{j=1}^i f(x_1 - j)) = f(x_1)] > \frac{1 + \delta}{2}$$

Hence we have a got a circuit C_f satisfying the claim in the lemma. Now, we handle the second case. Let x_1 be such that $\left| \Gamma(x_1, \otimes_{j=1}^i f(x_1 - j)) \right| > \delta^{k-1} + (k-1)\eta$. We take $x_1, \otimes_{j=1}^i f(x_1 - j)$ and the sign of $\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))$ (call it α) as advice (and this is at most $n + m$ bits) and define the circuit C^0 as follows.

$$C^0(\bar{y}, \otimes_{j=1}^i \bar{f}(\bar{y} - j)) = (-1)^\alpha C(x_1 \circ \bar{y}, \otimes_{j=1}^i f(x_1 - j) \bar{f}(\bar{y} - j))$$

By definition and the previous assumptions, we get the following

$$\mathbb{E}_{\bar{y} \in U_{(k-1)n}} \bar{f}(\bar{y}) C^0(\bar{y}, \otimes_{j=1}^i \bar{f}(\bar{y} - j)) > \delta^{k-1} + (k-1)\eta$$

Note that the above equation is same as (10) except circuit C has been replaced by C^0 and the input has changed from a k -tuple in $\{0, 1\}^n$ to a $k-1$ -tuple. Hence, this can be handled in an inductive way and the induction can go for at most $k-1$ steps. Further, each descent step in the induction can require at most $n+m$ bits of advice. In the step where we apply Lemma 3.5, we require at most $\frac{4nm}{\eta^2} + \log\left(\frac{4n}{\eta^2}\right) + 1$ bits of advice⁶. So, from T_2 , with at most $(k-1)(m+n) + \frac{4nk^2m^3}{\epsilon^2} + \log\left(\frac{4nk^2m^2}{\epsilon^2}\right) + 1$ bits of advice, we can get a circuit $C_f : \{0, 1\}^n \times \{0, 1\}^i$ such that

$$Pr_{x_1 \in U_n}[C_f(x_1, \otimes_{j=1}^i f(x_1 - j)) = f(x_1)] \geq \frac{1 + \delta}{2}$$

Finally accounting for the advice to use Lemma 2.5, we get that the total amount of advice required to get C_f from the circuit T in the hypothesis is $(k-1)(m+n) + \frac{4nk^2m^3}{\epsilon^2} + \log\left(\frac{4nk^2m^2}{\epsilon^2}\right) + 2 + m + \log m + 3 \leq \frac{6nk^2m^3}{\epsilon^2}$. ■

4 Conclusion

All the three extractor constructions described in this paper apply to sources of constant entropy rate, which could be pushed to entropy about $N/\text{poly} \log N$. A result of Viola [25] implies that it is impossible to extract from sources of entropy $N^{.99}$ if the extractor is such that each bit of the output can be computed by looking only at $N^{o(1)}$ bits of the input and seed length is $N^{o(1)}$. Since our construction is such that every bit of the output can be computed by looking at only $\text{poly} \log N$ bits of the input, significant improvements in the entropy rate can only come from rather different constructions.

It remains an interesting open question to improve the output length, and match the performance of other constructions which do not use complexity-theoretic tools in the analysis. Perhaps it is possible to use advice in a much more efficient way than we do.

⁶Note that η does not change for every step and is the same $\eta = \frac{\epsilon}{km}$ that it was set to in the beginning. The only extra condition we need for applying Lemma 3.5 is that $m \geq kn$ which shall definitely continue to hold as k decreases

References

- [1] Yonatan Aumann, Yan Zong Ding, and Michael O. Rabin. Everlasting security in the bounded storage model. *IEEE Transactions on Information Theory*, 48(6):1668–1680, 2002.
- [2] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudorandom bits. *SIAM Journal on Computing*, 13(4):850–864, 1984. Preliminary version in *Proc. of FOCS’82*.
- [3] Stefan Dziembowski and Ueli Maurer. Optimal randomizer efficiency in the bounded-storage model. *Journal of Cryptology*, 17(1):5–26, 2004. Preliminary version in *Proc. of STOC’02*.
- [4] O. Goldreich and L. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the 21st ACM Symposium on Theory of Computing*, pages 25–32, 1989.
- [5] O. Goldreich, N. Nisan, and A. Wigderson. On Yao’s XOR lemma. Technical Report TR95-50, Electronic Colloquium on Computational Complexity, 1995.
- [6] Alexander Healy. Randomness Efficient Sampling within NC^1 . *Computational Complexity*, 17(1):3–37, 2008.
- [7] Russell Impagliazzo, Ragesh Jaiswal, Valentine Kabanets, and Avi Wigderson. Direct product theorems: Simplified, Optimized and Derandomized. In *Proceedings of the 40th ACM Symposium on Theory of Computing*, pages 579–588, 2008.
- [8] Russell Impagliazzo and Avi Wigderson. $P = BPP$ unless E has sub-exponential circuits. In *Proceedings of the 29th ACM Symposium on Theory of Computing*, pages 220–229, 1997.
- [9] Leonid Levin. One-way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.
- [10] Chi-Jen Lu. Encryption against storage-bounded adversaries from on-line strong extractors. *Journal of Cryptology*, 17(1):27–42, 2004.
- [11] Ueli M. Maurer. Conditionally-perfect secrecy and a provably-secure randomized cipher. *J. Cryptology*, 5(1):53–66, 1992.
- [12] N. Nisan. Extracting randomness: How and why. In *Proceedings of the 11th IEEE Conference on Computational Complexity*, pages 44–58, 1996.
- [13] N. Nisan and A. Ta-Shma. Extracting randomness : A survey and new constructions. *Journal of Computer and System Sciences*, 1998. To appear. Preliminary versions in [12, 21].
- [14] N. Nisan and D. Zuckerman. More deterministic simulation in Logspace. In *Proceedings of the 25th ACM Symposium on Theory of Computing*, pages 235–244, 1993.
- [15] Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49:149–167, 1994. Preliminary version in *Proc. of FOCS’88*.
- [16] Noam Nisan and David Zuckerman. Randomness is linear in space. *Journal of Computer and System Sciences*, 52(1):43–52, 1996. Preliminary version in *Proc. of STOC’93*.
- [17] Omer Reingold, Salil Vadhan, and Avi Wigderson. Entropy waves, the zig-zag graph product, and new constant-degree expanders. *Annals of Mathematics*, 155(1):157–187, 2002.

- [18] Ronen Shaltiel. Recent developments in extractors. *Bulletin of the European Association for Theoretical Computer Science*, 77:67–95, 2002.
- [19] Ronen Shaltiel and Christopher Umans. Simple extractors for all min-entropies and a new pseudorandom generator. *Journal of the ACM*, 52(2):172–216, 2005.
- [20] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the XOR lemma. *Journal of Computer and System Sciences*, 62(2):236–266, 2001.
- [21] A. Ta-Shma. On extracting randomness from weak random sources. In *Proceedings of the 28th ACM Symposium on Theory of Computing*, pages 276–285, 1996.
- [22] A. Ta-Shma, D. Zuckerman, and S. Safra. Extractors from Reed-Muller codes. Technical Report TR01-036, Electronic Colloquium on Computational Complexity, 2001.
- [23] Luca Trevisan. Extractors and pseudorandom generators. *Journal of the ACM*, 48(4):860–879, 2001.
- [24] Christopher Umans. Pseudo-random generators for all hardnesses. *Journal of Computer and System Sciences*, 2(67):419–440, 2003.
- [25] Emanuele Viola. The complexity of constructing pseudorandom generators from hard functions. *Computational Complexity*, 13(3-4):147–188, 2004.
- [26] David Xiao and Avi Wigderson. A randomness-efficient sampler for matrix-valued functions and applications. In *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, pages 397–406, 2005.
- [27] Andrew C. Yao. Theory and applications of trapdoor functions. In *Proceedings of the 23th IEEE Symposium on Foundations of Computer Science*, pages 80–91, 1982.
- [28] Marius Zimand. Simple extractors via constructions of cryptographic pseudo-random generators. In *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming*, pages 115–127. LNCS 3580, Springer-Verlag, 2005.
- [29] David Zuckerman. General weak random sources. In *Proceedings of the 31st IEEE Symposium on Foundations of Computer Science*, pages 534–543, 1990.

A Extractor from direct product theorem

Description of the construction

The extractor $Ext : \{0, 1\}^{n2^n} \times \{0, 1\}^{2kn} \rightarrow \{0, 1\}^m$ is defined as follows. The weak random source is treated as a truth table of a function $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$. The seed is partitioned into two chunks of length kn each - call them \bar{x} and \bar{z} . \bar{x} is further partitioned into chunks of length n - call them $(x_1, x_2, x_3, \dots, x_k)$. Let $\bar{f}(\bar{x} + j)$ denote the tuple $(f(x_1 + j) \circ \dots \circ f(x_k + j))$. The i^{th} bit of the output is given by $\langle \bar{f}(\bar{x} + i - 1), \bar{z} \rangle$ where $\langle a, b \rangle$ denotes the inner product of a and b modulo 2. In terminology of section 2, $N = n2^n$ with $l = n$. Further, $g(k, n) = h(k, n) = kn$. As in the previous section, Exp simply partitions a string of length kn into k chunks of length n while Com computes the inner product of two strings.

Construction by Zimand in [28]

We note that the second construction in [28] is exactly the same construction we have described above. Our improved analysis can be used in the first construction in [28] as well but that would make the proof notationally cumbersome and we would also need to modify the results from [7] to this case. Hence for conceptual clarity, we just apply the analysis to the second construction.

Main theorem and proof of correctness

Theorem A.1 *The function $Ext : \{0, 1\}^{n2^n} \times \{0, 1\}^{2kn} \rightarrow \{0, 1\}^m$ is a $(\gamma n2^n, 2\epsilon + \frac{k^2}{2^n})$ strong extractor for a (constant) $\gamma > 0$, $\epsilon > 2^{-\frac{n}{7}}$, $m = O(\frac{\epsilon^{\frac{2}{3}} 2^{\frac{n}{3}}}{n})$ and seed length $2kn = O(\frac{n}{\gamma} \log \frac{m}{\epsilon})$,*

The following is an immediate corollary of the above theorem with parameters of interest.

Corollary A.2 *The function $Ext : \{0, 1\}^{n2^n} \times \{0, 1\}^{2kn} \rightarrow \{0, 1\}^m$ is a $(\gamma n2^n, 2\epsilon)$ strong extractor for a (constant) $\gamma > 0$, $2\epsilon = 2^{-n^{\frac{1}{4}}}$, $m = 2^{\frac{n}{3} - \sqrt{n}}$ and seed length $2kn = O(\frac{n^2}{\gamma})$.*

We first make the following simple observation. Let U_{nk} denote the uniform distribution over k -subsets of $\{0, 1\}^n$ and $[n^k]$ denote the set of all k -subsets of $\{0, 1\}^n$. By k -subsets of $\{0, 1\}^n$, we mean a subset of $\{0, 1\}^n$ of size k .

Observation A.3 *Let Ext_1 be the same as Ext except that the seed is drawn from the distribution $U_{nk} \times U_{kn}$ rather than U_{2kn} i.e. the input chunk is uniformly chosen k -subset rather than k -tuple. If Ext_1 is a strong $(\gamma 2^n, 2\epsilon)$ extractor, then Ext is a strong $(\gamma 2^n, 2\epsilon + \frac{k^2}{2^n})$ extractor with all other parameters remaining the same as Ext .*

Proof: Note that if we pick up k elements uniformly at random from a domain of size 2^n , then the probability that there is a collision is at most $\frac{k^2}{2^n}$. So, the statistical difference between the distributions of Ext and Ext_1 is at most $\frac{k^2}{2^n}$ and hence the observation follows. ■

Hence, from now on we shall assume that the seed is drawn from the distribution $U_{nk} \times U_{kn}$. We now state the main technical lemma of this section and show how Theorem A.1 immediately follows from it. Subsequently, we develop the machinery to prove the lemma.

Lemma A.4 *Let $T : \{0, 1\}^{m+kn} \rightarrow \{0, 1\}$ and $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$ such that (6) holds. Also, $0 < \delta \leq \frac{1}{8}$ be such that $\frac{2\epsilon^2}{m^2} \geq \exp(\frac{-\delta k}{D})$ where $D \geq 10^6$ is a constant and $\frac{2^n}{k} \geq \log(\frac{16}{\delta})$. Then for some $i < m$, using $\frac{14m^2 i \ln(\frac{1}{\delta}) kn}{\epsilon^2}$ bits of advice, we can get a circuit C_f such that*

$$Pr_{x \in U_n}[C_f(x, \otimes_{j=1}^i f(x-j)) = f(x)] \geq 1 - \frac{\delta}{2} \quad (14)$$

Proof: [of Theorem A.1] In light of Observation 2.2, we note that it is sufficient to prove that for any statistical test $T : \{0, 1\}^{m+kn} \rightarrow \{0, 1\}$, the number of functions f satisfying (6) is at most $\epsilon 2^{\gamma N} = \epsilon 2^{\gamma m 2^n}$. We set $\delta = \min\{\frac{1}{8}, \frac{\gamma}{2}\}$. Putting $k = \frac{2D \ln \frac{m}{\epsilon}}{\delta} = O(\frac{n}{\gamma})$ and δ as stated earlier (we consider the case when γ and hence δ are constants and n is sufficiently large), clearly satisfies the conditions for applying Lemma A.4. Hence for some $i < m$, with $\frac{14m^2 i \ln(\frac{1}{\delta}) kn}{\epsilon^2}$ bits of advice, we

can get a circuit C_f satisfying (14). Note that $i < m$ and hence putting the values of k, δ , we get that we need at most $\frac{C \ln \frac{1}{\gamma} 2^n}{\gamma^n}$ bits of advice (where C is a constant), are required to get C_f . Using Lemma 2.3, we can say that the total number of f satisfying (6) is at most (again the inequality assumes that γ is a constant and n is sufficiently large)

$$2^{\frac{C \ln \frac{1}{\gamma} 2^n}{\gamma^n} + mn + \frac{\delta}{2} n 2^n + H(\frac{\delta}{2}) 2^n} \leq 2^{\frac{\gamma n 2^n}{4}} 2^{\frac{C \ln \frac{1}{\gamma} 2^n}{\gamma^n} + mn + H(\frac{\delta}{2}) 2^n} \leq 2^{\frac{\gamma n 2^n}{2}} \leq 2^{-\frac{n}{7}} 2^{\gamma n 2^n}$$

The above inequality clearly proves Ext is an extractor. ■

We now develop the machinery for proving Lemma A.4. Using Lemma 2.5, and subsequently replacing i by $m - 1 - i$ and changing the first k arguments from x_j to $x_j + i$ for $1 \leq j \leq k$ (as done in the last section), we can assume that we have a circuit C which satisfies (15) for some $i < m$.

$$Pr_{\bar{x} \in U_{n,k}} Pr_{\bar{z} \in U_{nk}} [C(\bar{x}, \bar{z}, \otimes_{j=1}^i \langle \bar{f}(\bar{x} - j), \bar{z} \rangle) = \langle \bar{f}(\bar{x}), \bar{z} \rangle] > \frac{1}{2} + \frac{\epsilon}{m} \quad (15)$$

We now show how to change the ‘Hadamard code’ predictor to the ‘direct product’ predictor. More specifically, we prove the following lemma (Almost the same lemma is proven in [7], but we can avoid some of the complications because of being in the information-theoretic setting).

Lemma A.5 *Let C be a circuit such that it satisfies (15). Then using $2 \log \frac{m}{\epsilon} + 1$ bits of advice, we can get a circuit C_1 such that*

$$Pr_{\bar{x} \in U_{n,k}} [C_1(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j)) = \bar{f}(\bar{x})] > \frac{2\epsilon^2}{m^2} \quad (16)$$

Proof: We first construct a randomized circuit C_2 such that it satisfies (16). We then derandomize the circuit using advice bits. For $h \in \{0, 1\}^{nk}$, let us define $\lambda_{\bar{x}h}$ as follows.

$$\lambda_{\bar{x}h} = Pr_r [\langle h, r \rangle = C(\bar{x}, r, \otimes_{j=0}^i \langle \bar{f}(\bar{x} - j), r \rangle)] - \frac{1}{2}$$

Circuit C_2 on input $(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j))$ computes $\lambda_{\bar{x}h}$ for all $h \in \{0, 1\}^{nk}$ ⁷. Subsequently, it outputs h with probability $\frac{\lambda_{\bar{x}h}^2}{\sum_{h' \in \{0, 1\}^{nk}} \lambda_{\bar{x}h'}^2}$. We first claim that over the choices of \bar{x} and its internal randomness, C_2 has good probability of success in predicting $\bar{f}(\bar{x})$.

Claim A.6 *Let \bar{r}_1 represent the internal randomness of the circuit C_2 as described above. Then, the following holds*

$$Pr_{\bar{x}, \bar{r}_1} [C_2(\bar{x}, \bar{r}_1, \otimes_{j=1}^i \bar{f}(\bar{x} - j)) = \bar{f}(\bar{x})] > \frac{4\epsilon^2}{m^2} \quad (17)$$

Proof: Let us consider the function $\phi_{\bar{x}} : \{0, 1\}^{nk} \rightarrow \{0, 1\}$ which is defined (for $r \in \{0, 1\}^{kn}$) as

$$\phi_{\bar{x}}(r) = C(\bar{x}, r, \otimes_{j=1}^i \langle \bar{f}(\bar{x} - j), r \rangle)$$

We observe that $2\lambda_{\bar{x}h}$ is the fourier coefficient for $\phi_{\bar{x}}$ corresponding to h . By Parseval’s identity, we get that $4 \sum_{h' \in \{0, 1\}^{nk}} \lambda_{\bar{x}h'}^2 = 1$. Hence, the probability that a particular h is the output of C_2 is

⁷This is a simplification from [7]. Since we are in the information-theoretic setting, we can compute $\lambda_{\bar{x}h}$ by simply going over h and r individually. In the computational setting, even after using Goldreich-Levin theorem[4], one can only calculate approximations to $\lambda_{\bar{x}h}$ with high probability

$4\lambda_{\bar{x}h}^2$. In particular, the probability of success i.e. C_2 outputs $\bar{f}(\bar{x})$ is $4\lambda_{\bar{x}\bar{f}(\bar{x})}^2$. So, we get that over the choices of \bar{x} and the internal randomness of C_2 , the probability of success is $\mathbb{E}_{\bar{x}}4\lambda_{\bar{x}\bar{f}(\bar{x})}^2$. Also we note that by definition of C , $\mathbb{E}_{\bar{x}}\lambda_{\bar{x}\bar{f}(\bar{x})} > \frac{\epsilon}{m}$. Hence, by Jensen's inequality, we get that probability of success is

$$\mathbb{E}_{\bar{x}}4\lambda_{\bar{x}\bar{f}(\bar{x})}^2 \geq 4(\mathbb{E}_{\bar{x}}\lambda_{\bar{x}\bar{f}(\bar{x})})^2 > \frac{4\epsilon^2}{m^2}$$

By a Markov argument, we get that at least for $\frac{2\epsilon^2}{m^2}$ fraction of choices of the internal randomness \bar{r}_1 , we have $Pr_{\bar{x}}[C_2(\bar{x}, \bar{r}_1, \otimes_{j=1}^i \bar{f}(\bar{x} - j)) = \bar{f}(\bar{x})] > \frac{2\epsilon^2}{m^2}$. Hence, using $\lceil 2 \log \frac{m}{\epsilon} \rceil$ bits of advice, we can get such a 'good' \bar{r}_1 . We fix such a r_1 in C_2 to get the circuit C_1 . ■

We now state the following lemma and note that it immediately implies the main lemma.

Lemma A.7 *Let circuit C_1 satisfy (16) and $\delta < \frac{1}{8}$ be such that $\frac{2\epsilon^2}{m^2} \geq \exp\left(\frac{-\delta k}{D}\right)$ where $D \geq 10^6$ is a constant and $\frac{2^n}{k} \geq \log\left(\frac{16}{\delta}\right)$. Then using $\frac{13m^2 i \ln(\frac{1}{\delta})kn}{\epsilon^2}$ bits of advice, we can get a circuit C_f such that C_f satisfies*

$$Pr_{x \in U_n}[C_f(x, \otimes_{j=1}^i f(x - i)) = f(x)] \geq 1 - \frac{\delta}{2} \quad (18)$$

Proof: [of Lemma A.4] Let $T : \{0, 1\}^{m+kn} \rightarrow \{0, 1\}$ and $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$ such that (6) holds. Then from Lemma 2.5, using $m + \log m + 3$ bits of advice, we can get a circuit C satisfying (15). Further, using Lemma A.5, we can get a circuit C_1 satisfying (16) using $2 \log \frac{m}{\epsilon} + 1$ bits of advice. Subsequently, applying Lemma A.7, we get a circuit C_f satisfying (18) using $\frac{13m^2 i \ln(\frac{1}{\delta})kn}{\epsilon^2}$ bits of advice. Hence in all we require $\frac{13m^2 i \ln(\frac{1}{\delta})kn}{\epsilon^2} + m + \log m + 2 \log \frac{m}{\epsilon} + 4 \leq \frac{14m^2 i \ln(\frac{1}{\delta})kn}{\epsilon^2}$ bits of advice. ■

In order to prove lemma A.7, we will need some definitions (which are there in [7] in almost the same form) which we give below and then we state the relevant results from [7]. The results in [7] are in the context of the direct product theorem i.e. hypothesis is of the form that for some very small quantity ϵ , there is a circuit C_{dp} such that the following is true.

$$Pr_{\bar{x} \in U_{n^k}}[C_{dp}(\bar{x}) = \bar{f}(x)] > \epsilon$$

Direct product theorems say that with "small amount" of advice, one can construct a circuit C_s from C_{dp} such that for some small $\delta > 0$

$$Pr_{x \in U_n}[C_s(x) = f(x)] \geq 1 - \delta$$

As can be seen, the difference is that our hypothesis C_{dp} is of the following form

$$Pr_{\bar{x} \in U_{n^k}}[C_{dp}(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j)) = \bar{f}(x)] > \epsilon$$

and we would like to construct a circuit C_s which satisfies

$$Pr_{x \in U_n}[C_s(x, \otimes_{j=1}^i f(x - j)) = f(x)] \geq 1 - \delta$$

However, even though direct product theorems are a special case of the one we are treating here (correspond to $i = 0$), the algorithms and their proofs are the same in our setting except one

has to keep track of the extra information $\otimes \bar{f}(\bar{x} - j)$ and has to keep adding relevant advice for this auxillary input as and when required. To give an instance, if there is a place in the direct product theorem where one of the elements x_i in the tuple \bar{x} is fixed to some value, in our context it will translate to fixing x_i as well as fixing $\otimes_{j=1}^i f(x - j)$ which shall require additional advice. Apart from that, the theorems translate easily from one setting to another. In particular, the theorems showing “structural properties” (an instance of structural properties will be the proofs of correctness of various algorithms) of direct product and their proofs remain unaltered when switching between the two settings. Since, this section just uses some structural theorems from [7], we state the relevant results and refer the reader to [7] for the proofs.

For rest of the proof, we put $\epsilon_1 = \frac{2\epsilon^2}{m^2}$. Also all the definitions and the results in this section which implicitly or explicitly refer to the circuit C_1 assume that C_1 satisfies (16).

Definition A.8 *Circuit C_1 is said to be correct at \bar{x} if $C_1(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j)) = \bar{f}(\bar{x})$*

As has been noted earlier, \bar{x} represents a set of size k . So, we may sometimes refer to \bar{x} as a set (of size k). Also, any set $A \in \{0, 1\}^n$ of size k can be used as \bar{x} in (16). Further, in all subsequent definitions, whenever we consider a pair (A_1, A_2) , then $A_2 \in [n^k]$ i.e. A_2 is a subset of size k in $\{0, 1\}^n$ and A_1 is a subset of A_2 of size $\frac{k}{2}$ (assume w.l.o.g that k is even)

Definition A.9 *The pair (A_1, A_2) is said to be good if C_1 is correct at A_2 ⁸ and for Γ defined as $\Gamma = \{\bar{y} | \bar{y} \in [n^k] \text{ and } A_1 \subset \bar{y}\}$*

$$Pr_{\bar{y} \in \Gamma} [C_1 \text{ is correct at } \bar{y}] \geq \frac{\epsilon_1}{2}$$

Note that the property of a pair being good is fundamentally the property of the first argument i.e. A_1 rather than the pair (A_1, A_2) . However, the results in [7] have been proved with this terminology and hence it will be useful for us to stay with the same to straightaway plug in the results from there. We introduce one more notation now. Let \bar{x} be any k -subset and A_0 be any subset (of any size) of \bar{x} . Then $C_1(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j))_{A_0}$ represents the output of C_1 (in order) at the coordinates corresponding to A_0 .

Definition A.10 *Let A_1 be a $\frac{k}{2}$ -subset of size $\{0, 1\}^n$. $\bar{f}(A_1)$ represents the $\frac{k}{2}$ -tuple consisting of values of $f(y)$ for $y \in A_1$. A k -subset \bar{x} is said to be consistent with a $\frac{k}{2}$ -set A_1 if $A_1 \subset \bar{x}$ and $C_1(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j))_{A_1} = \bar{f}(A_1)$*

Definition A.11 *For any \bar{x} which is a k -subset of $\{0, 1\}^n$, $Inc(\bar{x})$ consists of all $y \in \{0, 1\}^n$ such that $y \in \bar{x}$ and $C_1(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j))_y \neq f(y)$. $Err(\bar{x})$ is defined as $\frac{|Inc(\bar{x})|}{k}$.*

Definition A.12 *A pair (A_1, A_2) is called α -excellent if (A_1, A_2) is good and $\mathbb{E}_{\bar{y} \in \Gamma} Err(\bar{y}) \leq \alpha$ where Γ consists of all k -subsets \bar{y} which are consistent with A_1 .*

Reconstruction algorithm

The way we have made the above definitions, it is not apriori clear if given circuit C_1 , there exists even one pair (A_1, A_2) which is good or α -excellent. The following lemma can be obtained by

⁸In [7], a pair (A_1, A_2) is defined to be correct rather than A_2 but since having the latter suffices, we go for what we believe is a more intuitive definition

plugging the parameters for the independent sampler from Lemma 2.8 in Lemma 3.7⁹ in [7] and putting $\alpha = \frac{\delta}{32}$ and $\epsilon' = \frac{\epsilon_1}{2}$. This lemma asserts that not only can you find $\frac{\delta}{32}$ -excellent pairs but there are a significant number of them. In all the subsequent results in this section, we are always referring to the circuit C_1 satisfying (16).

Lemma A.13 *If for every $1 \geq \mu > \frac{\delta}{64}$, $\mu \exp\left(\frac{-\mu k}{24}\right) \leq \frac{\epsilon_1^2 \delta}{1024}$, then a random pair (A_1, A_2) is $\frac{\delta}{32}$ -excellent with probability at least $\frac{\epsilon_1}{4}$*

Observe that without loss of generality, we can assume that $\frac{1}{10} \geq \epsilon_1^{10}$. With this, the following corollary is immediate.

Corollary A.14 *There exist constants D and F and $F \geq \frac{D}{2}$ (any $D \geq 10^6$ suffices) such that if $\epsilon_1 \geq \exp\left(-\frac{\delta k}{D}\right)$, then we can get a $\frac{\delta}{32}$ excellent pair (A_1, A_2) with at most $3 + \frac{\delta k}{F}$ bits of advice.*

Proof: First of all, by assumption $\exp\left(\frac{\delta k}{D}\right) \geq 10$ and since $D \geq 10^6$, hence we can conclude that $\delta k \geq 10^6$. Elementary calculus tells us that the function $\phi(\mu) = \mu \exp\left(\frac{-\mu k}{24}\right)$ is non increasing in the interval $[\frac{\delta}{64}, 1]$. So it suffices to verify that $\phi\left(\frac{\delta}{64}\right) < \frac{\delta \exp\left(\frac{-2\delta k}{D}\right)}{1024}$. However, this again easily follows from the fact that δk and D are both at least 10^6 . So, we see that the hypothesis of Lemma A.13 is satisfied. Hence, if we pick a random k -subset A_2 of $\{0, 1\}^n$ and subsequently pick a random $\frac{k}{2}$ subset of A_2 (call it A_1), then (A_1, A_2) is $\frac{\delta}{32}$ excellent with probability $\geq \frac{\epsilon_1}{4}$. Therefore the process of choosing a $\frac{\delta}{32}$ excellent pair can be derandomized using $\log\left(\frac{4}{\epsilon_1}\right) \leq 2 + \lceil \frac{\delta k}{F} \rceil \leq 3 + \frac{\delta k}{F}$ bits of advice¹¹. ■

We can now assume that we have a $\frac{\delta}{32}$ -excellent pair (A_1, A_2) . We use the circuit C_1 to describe a circuit C_0 which on input $x \in \{0, 1\}^n$ and $\otimes_{j=1}^i f(x - j)$ computes $f(x)$. First of all, using $\frac{kn}{2}$ bits of advice, we hardwire the value of $\bar{f}(A_1)$ in C_0 . Using further $\frac{ikn}{2}$ bits of advice, we also hardwire the value of $\otimes_{j=1}^i \bar{f}(A_1 - j)$. On input $x \in \{0, 1\}^n$ and $\otimes_{j=1}^i f(x - j)$, C_0 does the following (Our procedure is somewhat different from the procedure outlined in [7] due to some technical reasons - in particular “fixing the randomness” argument is tricky in [7] because we are dealing with k -subsets and not k -tuples):

If $x \in A_1$, then it simply returns the value of $f(x)$. If not, C_0 chooses a set S of size $\frac{k}{2} - 1$ uniformly at random such that $S \cap A_1 = \emptyset$. Circuit C_0 now takes $\otimes_{j=1}^i \bar{f}(S - j)$ advice. If $x \in S$, then C_0 discards this step. Let us call this event a collision. If not, then let \bar{y} denote the k -set $A_1 \cup S \cup \{x\}$. Note that with all the advice taken so far as well as the input, C_0 has the value of the tuple $\otimes_{j=1}^k \bar{f}(\bar{y} - j)$. Now, C_0 computes $C_1(\bar{y}, \otimes_{j=1}^i \bar{f}(\bar{y} - j))$ and if C_1 is consistent with A_1 at \bar{y} , then C_0 returns $C_1(\bar{y}, \otimes_{j=1}^i \bar{f}(\bar{y} - j))_x$. Note that since C_0 has already taken $\bar{f}(A_1)$ as advice, it can check the consistency of any k -subset with A_1 . If C_1 is inconsistent at \bar{y} or C_0 had discarded this particular step because of collision, then it repeats the same procedure. If the total number of iterations exceeds $\frac{48 \ln \frac{1}{\delta}}{\epsilon_1}$, then it outputs some fixed answer (say all zeros). Note that by repeating the procedure, we mean that it picks another random set S rather than picking another $\frac{\delta}{32}$ excellent pair. The following lemma states that C_0 succeeds with high probability.

⁹both refer to the numbering in [7]

¹⁰If $\epsilon_1 > \frac{1}{10}$, then put $\epsilon_1 = \frac{1}{10}$ and observe that the equations will continue to hold good

¹¹The constant got changed from D to F because we took the logarithm in base 2

Lemma A.15 *Let \bar{r}_2 be the internal randomness of C_0 as described above with $\epsilon_1 > \exp\left(\frac{-\delta k}{D}\right)$, $\frac{2^n}{k} \geq \log\left(\frac{16}{\delta}\right)$ and $\delta < \frac{1}{8}$. Then,*

$$Pr_{\bar{r}_2, x \in U_n}[C_0(x, \bar{r}_2, \otimes_{j=1}^i f(x-i)) = f(x)] \geq 1 - \frac{\delta}{4} \quad (19)$$

Before proving Lemma A.15, we show how it implies Lemma A.7.

Proof: [of Lemma A.7] We remind ourselves that we have been given the circuit C_1 and we want to construct a C_f satisfying (18). To do this, we first compute the amount of advice that C_0 uses. By Corollary A.14, to get a $\frac{\delta}{32}$ excellent pair (A_1, A_2) , we require at most $3 + \frac{\delta k}{F}$ bits of advice. Further, to get the value of $\bar{f}(A_1)$ as well as $\otimes_{j=1}^i \bar{f}(A_1 - j)$, we require $\frac{(i+1)kn}{2}$ bits of advice. We now note that by a Markov argument, for at least $\frac{1}{2}$ of the choices of \bar{r}_2 in (19), the following holds

$$Pr_{x \in U_n}[C_0(x, \bar{r}_2, \otimes_{j=1}^i f(x-i)) = f(x)] \geq 1 - \frac{\delta}{2}$$

Hence, using 1 bit of advice, we can get such a \bar{r}_2 which satisfies the above equation. Let us call the resulting circuit (formed by fixing \bar{r}_2) as C_f . Note that each of the $\frac{48 \ln \frac{1}{\delta}}{\epsilon_1} = \frac{24m^2 \ln \frac{1}{\delta}}{\epsilon^2}$ iterations requires at most $\frac{ikn}{2}$ bits of advice (which is independent of the input x). This is because once we fix the randomness \bar{r}_2 , we fix the set S being chosen by C_0 in every particular iteration. And the only advice C_0 needs every iteration is the tuple $\otimes_{j=1}^i \bar{f}(S-j)$ for that particular iteration which is $\frac{ikn}{2}$ bits of information. Thus, fixing the randomness in C_0 as well as hardwiring all the advice, we get the circuit C_f in (18). Clearly, the total advice required to get the $\frac{\delta}{32}$ -excellent pair as well as fixing the randomness \bar{r}_2 and getting the related advice is at most $3 + \frac{\delta k}{F} + \frac{(i+1)kn}{2} + 1 + \frac{12m^2 i \ln(\frac{1}{\delta})kn}{\epsilon^2} \leq \frac{13m^2 i \ln(\frac{1}{\delta})kn}{\epsilon^2}$ ■

We now come back to the proof of Lemma A.15. In order to prove this, we again use some results from [7]. The first lemma states that given a pair (A_1, A_2) is $\frac{\delta}{32}$ -excellent (actually, just being good suffices), for most of the inputs $x \in \{0, 1\}^n$, a significant fraction of k -subsets containing A_1 and x are consistent. (This lemma can be obtained by plugging $\beta = \frac{\delta}{16}$ in the independent sampler from Lemma 2.8 and putting in Lemma 3.5 in [7])

Lemma A.16 *Suppose (A_1, A_2) is a $\frac{\delta}{32}$ -excellent pair and $\exp\left(-\frac{\delta k}{400}\right) \leq \frac{\epsilon_1}{8}$. Let $Cons_{A_1}(x)$ denote the set of $\bar{y} \in [n^k]$ such that \bar{y} contains both x and A_1 and \bar{y} is consistent with A_1 . Let $Sub(A_1, x)$ consists of all $\bar{y} \in [n^k]$ such that \bar{y} contains both x and A_1 . Then for at most $\frac{\delta}{16}$ fraction of $x \in \{0, 1\}^n$, $\frac{|Cons_{A_1}(x)|}{|Sub(A_1, x)|} \leq \frac{\epsilon_1}{8}$.*

The following corollary states that one can show that for most of the $x \in \{0, 1\}^n$, the circuit C_0 is unlikely to fail because it never managed to sample a consistent tuple.

Corollary A.17 *Suppose (A_1, A_2) is a good pair and ϵ_1, k, δ satisfy the relation in Lemma A.15, then for at least $1 - \frac{\delta}{16}$ of the $x \in \{0, 1\}^n$, with probability $1 - \frac{\delta}{8}$, circuit C_0 samples a consistent k -set containing A_1 and x in $\frac{48 \ln \frac{1}{\delta}}{\epsilon_1}$ steps.*

Proof: First of all, for any particular x , we bound the probability that in $\frac{48 \ln \frac{1}{\delta}}{\epsilon_1}$ steps, more than $\frac{16 \ln \frac{1}{\delta}}{\epsilon_1}$ steps were discarded because for each of these iterations, there was a collision. (Note that

S in the description of C_0 is chosen independently for each iteration and hence collision in distinct rounds is independent). The probability of collision in a particular round is at most $\frac{k}{2^n}$. Therefore, by Chernoff bound, the probability that there were collisions in more than $\frac{16 \ln \frac{1}{\delta}}{\epsilon_1}$ steps is at most $\exp(-\frac{2^n}{k}) \leq \frac{\delta}{16}$. Further, since ϵ_1, k, δ satisfy the relation in Lemma A.15 i.e. $\frac{1}{10} > \epsilon > \exp(-\frac{\delta k}{D})$ where $D \geq 10^6$, hence in particular, $\exp(-\frac{\delta k}{400}) \leq \frac{\epsilon_1}{8}$ (again this uses $\delta k > 10^6$). Therefore, by Lemma A.16, we get that for at least $1 - \frac{\delta}{16}$ of the $x \in \{0, 1\}^n$ (call this set of x 's as T), the fraction of consistent tuple containing x and A_1 among all the tuples containing x and A_1 is at least $\frac{\epsilon_1}{8}$. Now conditioning on the event that less than $\frac{16 \ln \frac{1}{\delta}}{\epsilon_1}$ steps were discarded i.e. least $\frac{32 \ln \frac{1}{\delta}}{\epsilon_1}$ k -subsets containing x and A_1 were sampled, we can say that for $x \in T$, the probability that we do not sample any consistent tuple in $\frac{32 \ln \frac{1}{\delta}}{\epsilon_1}$ steps is at most $\delta^4 \leq \frac{\delta}{16}$. So, for $x \in T$, the probability that we do not sample any consistent tuple containing x and A_1 is at most $\frac{\delta}{16} + \frac{\delta}{16} = \frac{\delta}{8}$. As mentioned before the density of T is at least $1 - \frac{\delta}{16}$. ■

The next lemma (obtained by plugging $\alpha = \frac{\delta}{32}, \beta = \frac{\delta}{16}$ and the value of independent sampler from Lemma 2.8 in Lemma 3.6 from [7]) says that for a random x , if C_0 samples a consistent k -set, it is likely to get the correct value using C_1 . More formally, as before let the set of k -sets containing A_1 and x which are consistent with A_1 be denoted by $Cons_{A_1}(x)$. Also, $\bar{y} \in Cons_{A_1}(x)$ is contained in the set $Corr(x)$ iff $C_1(\bar{y}, \otimes_{j=1}^i \bar{f}(\bar{y} - j))_x = f(x)$. Let $h(x) = \frac{|Corr(x)|}{|Cons_{A_1}(x)|}$.

Lemma A.18 *Let (A_1, A_2) be a $\frac{\delta}{32}$ excellent pair, $\exp(-\frac{\delta k}{400}) \leq \frac{\epsilon_1}{8}$ and $h(x)$ be as defined above. Then $\mathbb{E}_{x \in \{0,1\}^n} h(x) \leq \frac{\delta}{16}$.*

Combining the above lemmas, we get the proof of Lemma A.15.

Proof: [of Lemma A.15] The circuit C_0 can fail to compute $f(x)$ correctly for one of the two reasons. One is that it fails to sample a consistent k -subset containing A_1 and x in $t = \frac{48 \ln \frac{1}{\delta}}{\epsilon_1}$ iterations. Second is that given that a consistent k -subset was sampled, the answer produced by C_1 was wrong. We note that using corollary A.17 (note that all conditions for applying Corollary A.17 are satisfied), we can say that for at least $1 - \frac{\delta}{16}$ fraction of the x 's, we sample a consistent k -subset within $\frac{48 \ln \frac{1}{\delta}}{\epsilon_1}$ iterations with probability at least $1 - \frac{\delta}{8}$. Therefore, for a random x , the failure probability because C_f failed to sample a consistent set is at most $\frac{\delta}{16} + \frac{\delta}{8} = \frac{3\delta}{16}$. Note that conditioned on the event that a consistent k -subset was sampled for a random x , the probability of failure because C_1 computes a wrong value of $f(x)$ is exactly $\mathbb{E}_{x \in \{0,1\}^n} h(x)$. Also, note that in the proof of Lemma A.17, we had already confirmed that $\exp(-\frac{\delta k}{400}) \leq \frac{\epsilon_1}{8}$. Hence, we can apply Lemma A.18. Therefore, $\mathbb{E}_{x \in \{0,1\}^n} h(x) \leq \frac{\delta}{16}$. Hence, the total probability of failure over the random choices of x and internal coin tosses of C_0 is $\frac{\delta}{16} + \frac{3\delta}{16} = \frac{\delta}{4}$. ■

B Derandomization of construction

The reason that seed length was $O(n \log \frac{m}{\epsilon})$ rather than $O(n)$ in both the previous constructions is that the extractor picks up $O(\log \frac{m}{\epsilon})$ independent instances on which it evaluates the function f (represented by the weak random source). As evaluation of f on independent instances is just a way to get a 'hard' function from f (as done in direct product theorems), a natural alternative to get the same 'hardness' but with a smaller seed length is to use a derandomized version of the direct

product theorem. As we would like to have output length $2^{O(n)}$, hence we need to increase the hardness of our function from a constant to $2^{-\Omega(n)}$. Several known hardness amplification methods achieve this, in particular, Impagliazzo and Wigderson [8], Sudan, Trevisan and Vadhan [20] and Shaltiel and Umans [19, 24]. However, it is not clear how to interpret the results in [20, 19, 24] as derandomized direct products. Hence, we use the derandomized direct product theorem from [8]¹². We heavily use the machinery developed in [8]. The following definitions are from [8] which we need in order to define the construction of the extractor.

Definition B.1 [8] *A polynomial time computable function $G : \{0, 1\}^{n_1} \rightarrow (\{0, 1\}^n)^k$ is called a S -restrictable pseudorandom generator if there is a function $h : [k] \times \{0, 1\}^n \times \{0, 1\}^{n_1-n} \rightarrow \{0, 1\}^{n_1}$ such that the following hold. Let $(x_1, x_2, x_3, \dots, x_k)$ be the output of $G(h(i, x, \alpha))$*

- For uniformly chosen x and α and fixed i , $h(i, x, \alpha)$ is uniformly distributed in $\{0, 1\}^{n_1}$
- For a fixed i , x and α , $x_i = x$.
- For a fixed α , i and j , there exists a set $S_{\alpha j}$ and $|S_{\alpha j}| \leq S$ such that for any value of x , $x_j \in S_{\alpha j}$

We call h the “permuting function” for G .

Definition B.2 [8] *A polynomial time computable function $G : \{0, 1\}^{n_2} \rightarrow (\{0, 1\}^n)^k$ is called a (ρ, δ) hitting generator if the following holds: Let (X_1, X_2, \dots, X_k) be the output distribution of G when the input distribution is U_{n_2} . Let $H_1, H_2, \dots, H_k \subset \{0, 1\}^n$ be such that for any j , $|H_j| \geq \delta 2^n$. Then for $(x_1, x_2, x_3, \dots, x_k)$ drawn from the distribution (X_1, X_2, \dots, X_k)*

$$\Pr \left[|\{x_j : x_j \in H_j\}| < \frac{\delta k}{2} \right] \leq \rho$$

The following lemma was proven in [8] which states that we can get a pseudorandom generator meeting both definition B.1 and definition B.2 by meeting them individually and then taking a bitwise XOR. More formally, the following is shown

Lemma B.3 [8] *Let $G_1 : \{0, 1\}^{n_1} \rightarrow (\{0, 1\}^n)^k$ be a S -restrictable pseudorandom generator and $G_2 : \{0, 1\}^{n_2} \rightarrow (\{0, 1\}^n)^k$ be a (δ, ρ) hitting generator. Let the generator $G : \{0, 1\}^{n_1+n_2} \rightarrow (\{0, 1\}^n)^k$ be defined as: i^{th} bit of $G(x, y)$ ($|x| = n_1$ and $|y| = n_2$) is $G_1(x) \oplus G_2(y)$. Then G is S -restrictable and (δ, ρ) hitting.*

We now show how to meet the first and the second definitions. This has already been done in [8] but we do it here for the sake of completeness (and for some minor details we need). The following lemma (Lemma 18 in [8]) shows existence of a generator meeting Definition B.1.

Lemma B.4 *For any $\beta > 0$, n , k and $n_1 = \frac{2n}{\beta}$ there is a probabilistic polynomial time algorithm A such that with probability $1 - k^2 2^{-\Omega(\beta n)}$, it produces a description of $G : \{0, 1\}^{n_1} \rightarrow (\{0, 1\}^n)^k$ such that G is $2^{\beta n}$ -restrictable. Further A uses $O(n_1)$ bits of randomness and output of G can be computed in time polynomial in n given the description.*

¹²There are fundamental problems in using arbitrary derandomized direct product results and the result in [8] seems to be the only one which we can use

The following result which was shown by Healy in [6] (and implicitly present in the work by Xiao and Wigderson in [26]) is used to meet Definition B.2. It essentially states that a random walk on an expander graph satisfies the Definition B.2 with a ρ exponentially small in k . More specifically, the following is shown.

Lemma B.5 [6] *Let G be an expander graph over vertex set $\{0,1\}^n$ with second (normalized) eigenvalue $\lambda_2 < 1$ and degree d . Let v_1 be a randomly chosen vertex in $\{0,1\}^n$ and the walk $(v_1, v_2, v_3, \dots, v_k)$ is defined by picking v_i to be a random neighbor of v_{i-1} for $i > 1$. For $1 \leq i \leq k$, let $H_i \subseteq \{0,1\}^n$ such that $|H_i| \geq \delta 2^n$. Then $\Pr[|\{j | x_j \in H_j\}| \leq \frac{\delta k}{2}] \leq \exp(\Omega(-\delta^2(1 - \lambda_2)k))$*

It is clear that the above random walk takes $n + k \log d$ random bits in comparison to kn random bits had all the vertices been chosen uniformly randomly. In particular, if d is a large enough constant such that $\lambda_2 \leq \frac{1}{2}$, then the above random walk requires $n + O(k)$ random bits and the deviation bound in lemma B.5 can simply be written as $\exp(\Omega(-\delta^2 k))$. To use it in our construction, the expander graph should be fully explicit i.e. given a vertex, one can calculate its i^{th} neighbor in $\text{poly}(\log |V|)$ time where V is the vertex set. We can use any such construction - we use the one due to Reingold, Vadhan and Wigderson [17].

Lemma B.6 [17] *There exists a fully explicit constant degree (say d) graph on $\{0,1\}^n$ such that its second largest eigenvalue is bounded by $\frac{1}{2}$*

Hence, we get the following pseudorandom generator meeting definition B.2.

Lemma B.7 *There exists a $G : \{0,1\}^{n_2} \rightarrow (\{0,1\}^n)^k$ which is $(\delta, \exp(\frac{-\delta^2 k}{40}))$ hitting such that G is computable in time polynomial in n and k and $n_2 = n + O(k)$.*

The constant appearing in the expression i.e. $\frac{1}{40}$ is obtained by combining the constants one gets in Lemma B.5 and Lemma B.6. Also since the constant does not depend on δ , hence from now on, we shall remove the prefix $(\delta, \exp(\frac{-\delta^2 k}{40}))$ and just call such a G , a hitting generator (k shall be clear from the context). In fact, for this section, we define hitting generators to be ones which are $(\delta, \exp(\frac{-\delta^2 k}{40}))$ hitting. By applying Lemma B.3 to combine the pseudorandom generators from lemma B.4 and lemma B.7, we get the following lemma which gives a pseudorandom generator meeting both Definitions B.1 and B.2.

Lemma B.8 *There exists a probabilistic polynomial time algorithm A_1 such that for any $\beta > 0$, n , k and $r = O(\frac{n}{\beta} + k)$ which uses $\frac{Cn}{\beta}$ random bits (for some $C > 0$) and with probability $1 - k^2 2^{-\Omega(\beta n)}$ produces description of a generator $G : \{0,1\}^r \rightarrow (\{0,1\}^n)^k$ such that it is $2^{\beta n}$ -restrictable and hitting. Further G can be computed in time polynomial in n and k given the output of A_1 .*

Remark B.9 Without loss of generality, we can assume that irrespective of whether or not the output of A_1 is a description of $2^{\beta n}$ -restrictable and hitting generator, it is always description of a function $G : \{0,1\}^r \rightarrow (\{0,1\}^n)^k$.

Description of extractor

The extractor $Ext : \{0,1\}^{2^n} \times \{0,1\}^t \rightarrow \{0,1\}^m$ is defined as follows. Here $t = \frac{Cn}{\beta} + r + k$ where the symbols have the same meaning as in Lemma B.8. The source X is treated as truth table of a function $f : \{0,1\}^n \rightarrow \{0,1\}$. The seed is broken into three chunks \bar{w} , \bar{X}_1 and \bar{z} where

$|\bar{w}| = \frac{Cn}{\beta}$, $|\bar{X}_1| = r$ and $|\bar{z}| = k$. First the algorithm A_1 in lemma B.8 takes as input \bar{w} and produces description of $G : \{0, 1\}^r \rightarrow (\{0, 1\}^n)^k$ (Note that G is not necessarily a $2^{\beta n}$ restrictable and hitting generator). Subsequently, G takes as input \bar{X}_1 and produces as output x_1, x_2, \dots, x_k such that each $x_i \in \{0, 1\}^n$. Let \bar{x} denote the concatenation of x_i 's. Let $\bar{f}(\bar{x} + j)$ denote the tuple $(f(x_1 + j) \circ \dots \circ f(x_k + j))$. The i^{th} bit of the output is given by $\langle \bar{f}(\bar{x} + i - 1), \bar{z} \rangle$ where $\langle a, b \rangle$ denotes the inner product of a and b modulo 2. In terminology of section 2, $N = 2^n$ with $l = 1$. Further, $g(k, n) = r$ and $h(k, n) = k$. Note that $\bar{X}_1 \in \{0, 1\}^r$ while $\bar{x} \in (\{0, 1\}^n)^k$. Also, note that $r = O\left(\frac{n}{\beta} + k\right)$ and hence $t = O\left(\frac{n}{\beta} + k\right)$. As we shall later see, β shall be fixed to a constant and hence r will just be a function of n and k . In particular, r shall be $O(n+k)$. Also, the function Exp is the same function as G and Com is simply taking inner product of two strings. Note that there is one deviation from the description in Section 2 which is that $G \equiv Exp$ is defined probabilistically rather than being a fixed function.

Main theorem and proof of correctness

Theorem B.10 *For any $\frac{1}{6} > \beta > 0$, the function $Ext : \{0, 1\}^{2^n} \times \{0, 1\}^t \rightarrow \{0, 1\}^m$ is a $(\gamma 2^n, 2\epsilon + 2^{-\Omega(n)})$ strong extractor for a (constant) $\gamma > 0$, $\epsilon \geq 2^{-\frac{n}{7}}$, $m = \frac{\epsilon^{\frac{4}{5}} 2^{\frac{n(1-\beta)}{5}}}{n}$ and seed length $t = O\left(\frac{n}{\beta} + k\right)$ with $k = O\left(\frac{\log \frac{m}{\epsilon}}{\gamma^4}\right)$*

The following is an immediate corollary of theorem B.10 with parameters of interest.

Corollary B.11 *The function Ext as defined above is a $(\gamma 2^n, 2\epsilon)$ strong extractor for a (constant) $\gamma > 0$, $2\epsilon = 2^{-n^{\frac{1}{4}}}$, $m = 2^{\frac{19n}{100}}$ and seed length $t = O\left(\frac{n}{\gamma^4}\right)$.*

Instead of proving Theorem B.10, we prove the following lemma.

Lemma B.12 *For $\frac{1}{6} > \beta > 0$, let $G' : \{0, 1\}^{r'} \rightarrow (\{0, 1\}^n)^k$ be any $2^{\beta n}$ restrictable hitting generator and $Ext_1 : \{0, 1\}^{2^n} \times \{0, 1\}^{t_1} \rightarrow \{0, 1\}^m$ be defined in terms of G' exactly as the output of Ext was defined in terms of G . Then Ext_1 is a $(\gamma 2^n, 2\epsilon)$ strong extractor with exactly the same conditions on m and ϵ as in Theorem B.10 and seed length $t_1 = r' + k$ with $k = O\left(\frac{\log \frac{m}{\epsilon}}{\gamma^4}\right)$*

Remark B.13 When we call G' a $2^{\beta n}$ restrictable hitting generator, the hitting property is the same discussed previously i.e. for all $\delta > 0$, G' is $\left(\delta, \exp\left(-\frac{\delta^2 k}{40}\right)\right)$ hitting.

Before proving Lemma B.12, we first show how proving it suffices to prove Theorem B.10.

Proof: [of Theorem B.10] Let the event that A_1 does not produce a $2^{\beta n}$ restrictable hitting generator be called Γ . We see that probability of Γ is bounded by $k^2 2^{-\Omega(n)}$. Since $k = O\left(\frac{\log \frac{m}{\epsilon}}{\gamma^4}\right)$, we get that for constant $\gamma > 0$ and large enough n , the probability of Γ is bounded by $2^{-\Omega(n)}$. Conditioned on Γ not occurring, we can apply Lemma B.12 to get that the output of Ext is 2ϵ close to uniform. So, the total statistical distance of output of Ext from uniform is at most $2\epsilon + 2^{-\Omega(n)}$. Clearly, the total seed required is $r + k$ and the randomness used by A_1 which is $O\left(\frac{n}{\beta}\right)$. As r itself is $O\left(\frac{n}{\beta} + k\right)$, so the total seed length remains $O\left(\frac{n}{\beta} + k\right)$. ■

We now state the main technical lemma of this section and then show Lemma B.12 follows from it. Subsequently, the machinery to prove the lemma is developed.

Lemma B.14 *Let $T : \{0, 1\}^{m+t_1} \rightarrow \{0, 1\}$ and $f : \{0, 1\}^n \rightarrow \{0, 1\}$ such that (6) holds and $\frac{\epsilon^2}{m^2} \geq 3 \exp(\frac{-\delta^2 k}{40})$. Then for some $i < m$, using $O\left(\frac{nm^4(i+1)k2^{\beta n}}{\epsilon^4}\right)$ bits of advice, we can get a circuit C_f such that*

$$Pr_{x \in U_n}[C_f(x, \otimes_{j=1}^i f(x-j)) = f(x)] \geq 1 - \delta \quad (20)$$

Proof: [of Lemma B.12] In light of Observation 2.2, we note that it is sufficient to prove that for any statistical test $T : \{0, 1\}^{m+t_1} \rightarrow \{0, 1\}$, the number of functions f satisfying (6) is at most $\epsilon 2^{\gamma N}$. Let $\delta = \min\{10^{-3}, \frac{\gamma^2}{4}\}$. Also put $k = \Theta\left(\frac{\log \frac{m}{\gamma^4 \epsilon}}{\gamma^4}\right) = O\left(\frac{n}{\gamma^4}\right)$. With appropriate choice of constants inside the $\Theta(\cdot)$ notation, we see that $\frac{\epsilon^2}{m^2} \geq 3 \exp\left(\frac{-\delta^2 k}{40}\right)$. With this, we satisfy the conditions for applying Lemma B.14 and hence we can say that using $O\left(\frac{nm^4(i+1)k2^{\beta n}}{\epsilon^4}\right)$ bits of advice, we can get a circuit C_f satisfying (20). Using Fact 3.4, we immediately get that $\delta + H(\delta) \leq \frac{\gamma}{2}$. Note that the seed length is $t_1 = r' + k$ and with $k = O\left(\frac{\log \frac{m}{\gamma^4 \epsilon}}{\gamma^4}\right)$. Putting everything together now, we get that the total number of functions satisfying (6) is at most (we consider the case when $\gamma > 0$ is a constant and n is large enough integer).

$$2^{O\left(\frac{nm^4(i+1)k2^{\beta n}}{\epsilon^4}\right) + (\delta + H(\delta))2^n + m} \leq 2^{O\left(\frac{nm^5 k 2^{\beta n}}{\epsilon^4}\right) + (\delta + H(\delta))2^n + m} \leq 2^{O\left(\frac{2^n}{\gamma^4 n^3}\right) + \frac{\gamma}{2} 2^n + 2^{\frac{n}{5}}} \leq 2^{-\frac{n}{7}} 2^{\gamma 2^n} \leq \epsilon 2^{\gamma 2^n}$$

■

We now develop the machinery to prove Lemma B.14. Let $G'(n, k)$ denote the distribution of output of G' when its input is drawn from the uniform distribution over $\{0, 1\}^{r'}$. Using Lemma 2.5, and subsequently replacing i by $m - 1 - i$ and changing the first k arguments from x_j to $x_j + i$ for $1 \leq j \leq k$ (as done in the last section), we can assume that we have a circuit C which satisfies (21) for some $i < m$. Observe that the step in which we change the argument from x_j to $x_j + i$ can be done because a M -restrictable hitting distribution remains so even if each of the individual points in the support of the distribution is shifted by some constant vector.

$$Pr_{\bar{x} \in G'(n, k)} Pr_{\bar{z} \in U_k}[C(\bar{x}, \bar{z}, \otimes_{j=1}^i \langle \bar{f}(\bar{x} - j), \bar{z} \rangle) = \langle \bar{f}(\bar{x}), \bar{z} \rangle] > \frac{1}{2} + \frac{\epsilon}{m} \quad (21)$$

Note that a simple application of Lemma 2.5 would have resulted in having a circuit C with the first argument as $\bar{X}_1 \in \{0, 1\}^{r'}$. However, we can assume that the first argument is $\bar{x} \in (\{0, 1\}^n)^k$ being sampled from the distribution $G'(n, k)$ because this can be done by the circuit C on its own when given \bar{X}_1 . We now change the ‘Hadamard-code’ predictor to a ‘direct-product’ predictor exactly as we did in Lemma A.5 (the proof is exactly the same except that the distribution from which \bar{x} is sampled has changed. But that does not affect the proof. Hence the proof is not included here) and get the following result.

Lemma B.15 *Let C be a circuit such that it satisfies (21). Then using $2 \log \frac{m}{\epsilon} + 1$ bits of advice, we can get a circuit C_1 such that*

$$Pr_{\bar{x} \in G'(n, k)}[C_1(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x} - j)) = \bar{f}(\bar{x})] > \frac{2\epsilon^2}{m^2} \quad (22)$$

We shall now state the following lemma and then see how it implies the main lemma (Lemma B.14) follows from it.

Lemma B.16 *Let C_1 be a circuit which satisfies (22) such that $\frac{\epsilon^2}{m^2} \geq 3 \exp(\frac{-\delta^2 k}{40})$. Then using $O\left(\frac{nm^4(i+1)k2^{\beta n}}{\epsilon^4}\right)$ bits of advice, we can get a circuit C_f such that*

$$\Pr_{x \in U_n}[C_f(x, \otimes_{j=1}^i f(x-j)) = f(x)] \geq 1 - \delta \quad (23)$$

Proof: [of Lemma B.14] Let $T : \{0,1\}^{m+t_1} \rightarrow \{0,1\}$ and $f : \{0,1\}^n \rightarrow \{0,1\}$ such that (6) holds. Then from Lemma 2.5, using $m + \log m + 3$ bits of advice, we can get a circuit C satisfying (21). Further, using Lemma B.15, we can get a circuit C_1 satisfying (22) using $2 \log \frac{m}{\epsilon} + 1$ bits of advice. Finally, using Lemma B.16, we get a circuit C_f satisfying (23) using $O\left(\frac{nm^4(i+1)k2^{\beta n}}{\epsilon^4}\right)$ bits of advice. Hence in all, to get C_f from T , we require $O\left(\frac{nm^4(i+1)k2^{\beta n}}{\epsilon^4}\right) + m + \log m + 2 \log \frac{m}{\epsilon} + 4 \leq O\left(\frac{nm^4(i+1)k2^{\beta n}}{\epsilon^4}\right)$ bits of advice. ■

Reconstruction algorithm

In this part, we heavily use the results from [8]. As had been discussed in Section 1 as well as Appendix A, the proofs in direct product setting carry over to our setting just at the cost of increasing the advice. In particular, the correctness properties of various algorithms (which is what we termed “structural properties” in Section A) translates easily between the two settings. Hence, we just state the relevant results from [8] without proof. For rest of the proof, put $\epsilon_1 = \frac{2\epsilon^2}{m^2}$.

In order to prove Lemma B.16, we will need the following definition. Whenever we refer to circuit C_1 in this section, we always refer to C_1 which satisfies (22).

Definition B.17 *For $l \in [k]$ and $x \in (\{0,1\}^n)^k$, Circuit C_1 is said to be correct at (l, \bar{x}) if the l^{th} bit in the output of $C_1(\bar{x}, \otimes_{j=1}^i \bar{f}(\bar{x}-j))$ is $f(x_l)$ where $\bar{x} = (x_1, x_2, \dots, x_k)$. Note that the definition only makes sense when \bar{x} is in support of $G'(n, k)$.*

To give the final construction of circuit C_f in Lemma B.16, we proceed in 3 stages:

- From circuit C_1 , we construct a distribution of circuits such that on any large set, a random sample from the distribution does somewhat better than a random guess
- Repeated sampling from this distribution and subsequently taking majority to get another circuit C_0 which computes $f(x)$ for a random x w.h.p.
- Fix the randomness of the circuit C_0 to get C_f

Given circuit C_1 , we now describe construction of circuit C_2 which on input x and $\otimes_{j=1}^i f(x-j)$ tries to compute $f(x)$. (How C_2 relates to a distribution shall be evident shortly) More specifically, it picks $s \in [k]$ and $v \in \{0,1\}^{r'-n}$ uniformly at random. Let $(x_1, \dots, x_k) = \bar{x} = G'(h(s, x, v))$ where h is the “permuting function” defined in Definition B.1 corresponding to G' . C_2 then takes the values $\otimes_{j=0}^i f(x_t - j)$ for all $t \neq s$ as advice. Note that C_2 has now got enough information to compute C_1 on \bar{x} and check apart from the s^{th} position, at which of the positions C_1 is correct in predicting f . Let u denote the number of positions $q \neq s$ such that C_1 is correct at (q, \bar{x}) . Then with probability 2^{-u} , C_2 outputs $f(x_s)$ and otherwise outputs 0 and 1 with equal probability.

The following lemma (obtained by plugging $\rho = \frac{\delta}{2}$ and $q = 2^{-\frac{\delta^2 k}{40}}$ in Theorem 15 and Lemma 16 proved in [8]) states that on every set $H \subset \{0, 1\}^n$ of size at least $\delta 2^n$, the circuit C_2 described above does somewhat better than a random guess would have done. More specifically, we have the following

Lemma B.18 *Let $\delta > 0$, $H \subset \{0, 1\}^n$ be any set of size at least $\delta 2^n$ and U_H denote the uniform distribution on H . Then for the circuit C_2 as described above,*

$$\Pr_{x \in U_H}[C_2(x, \otimes_{j=1}^i f(x-j)) = f(x)] \geq \frac{1}{2} + \frac{\epsilon_1}{4} - \frac{3 \exp(-\frac{\delta^2 k}{40})}{4} - \frac{2^{-\frac{\delta k}{6}}}{2}$$

Corollary B.19 *If $\epsilon_1 \geq 6 \exp(-\frac{\delta^2 k}{40})$, then for every set $H \subset \{0, 1\}^n$ of size at least $\delta 2^n$, and the circuit C_2 described above,*

$$\Pr_{x \in U_H}[C_2(x, \otimes_{j=1}^i f(x-j)) = f(x)] \geq \frac{1}{2} + \frac{\epsilon_1}{24}$$

Observe that in the hypothesis of Lemma B.16, we indeed have that $\epsilon_1 \geq 6 \exp(-\frac{\delta^2 k}{40})$. So, Corollary B.19 applies to our case. Also note that the circuit C_2 described above involves making random choices. Let the corresponding distribution from which circuit C_2 is being sampled be called D . For $x \in \{0, 1\}^n$ define the quantity $Adv(x)$ as follows

$$Adv(x) = \Pr_{C_2 \in D}[C_2(x, \otimes_{j=1}^i f(x-j)) = f(x)] - \frac{1}{2}$$

Let us now define Y which represents the set of bad inputs for C_2 . More specifically, set Y is defined as follows

$$Y = \{x | x \in \{0, 1\}^n \text{ and } Adv(x) < \frac{\epsilon_1}{24}\}$$

By corollary B.19, we can say that $|Y| \leq \delta 2^n$. We now describe construction of C_0 . C_0 picks circuits $C'_1, \dots, C'_{\Theta(\frac{n}{\epsilon_1^2})}$ (with an appropriately large constant inside the $\Theta(\cdot)$ notation) from the distribution X independently and then outputs the majority of $C'_1(x, \otimes_{j=1}^k f(x-j)), \dots, C'_{\Theta(\frac{n}{\epsilon_1^2})}(x, \otimes_{j=1}^k f(x-j))$.

Observation B.20 *Let \bar{r}_2 denote the internal randomness of circuit C_0 described above. Then for $x \notin Y$*

$$\Pr_{\bar{r}_2}[C_0(x, \bar{r}_2, \otimes_{j=1}^i f(x-i)) \neq f(x)] < 2^{-2n}$$

Proof: Follows from Chernoff bound and definition of Y . ■

By the above observation and applying a union bound, we get that for $1 - 2^{-n}$ fraction of choices of internal randomness \bar{r}_2 , $C_0(x, \bar{r}_2, \otimes_{j=1}^i f(x-i)) = f(x)$ for all $x \notin Y$. Thus with at most $\lceil \log \left(\frac{1}{1-2^{-n}} \right) \rceil \leq 1$ bit of advice, we can fix the internal randomness so that if C_0 commits a mistake in computing $f(x)$, then $x \in Y$. Therefore, we get that after fixing the internal randomness and the corresponding advice (and calling the resulting circuit C_f)

$$\Pr_{x \in U_n}[C_f(x, \otimes_{j=1}^i f(x-j)) = f(x)] \geq 1 - \delta \tag{24}$$

Finally, we have the required construction to prove Lemma B.16.

Proof: [of Lemma B.16] We first note that in order to fix the internal randomness of C_0 (to get C_f) we need at most 1 bit of advice. Further, upon fixing the value of the internal randomness \bar{r}_2 , for any particular C'_q (i.e. the q^{th} circuit sampled by C_0 from distribution D), we have also fixed the value of s and v for C'_q . By definition of G being $2^{\beta n}$ restrictable, for any $t \neq s$, $\exists S_t \subset \{0, 1\}^n$ such that $x_t \in S_t$ $|S_t| \leq 2^{\beta n}$. This also implies that for any j and $t \neq s$, $\exists S_{tj} \subset \{0, 1\}^n$, $(x_t - j) \in S_{tj}$ where $|S_{tj}| \leq 2^{\beta n}$. For computing $C'_q(x, \otimes_{j=1}^i f(x - j))$, the only remaining information (other than what we have already accounted for) required is the value of $\otimes_{j=0}^i f(x_t - j)$ for all $t \neq s$. Hence with $(i + 1)(k - 1)2^{\beta n}$ bits of advice, we can hardwire the value of $\otimes_{j=0}^i f(x_t - j)$ in circuit C_f for any particular q . Therefore, to hardwire the information for computing C_f , we just need to hardwire the advice for every C'_q for all possible q . So, the total amount of advice required to get C_f is $O\left(\frac{n(i+1)(k-1)}{\epsilon_1^2}\right) 2^{\beta n} + 1 = O\left(\frac{n(i+1)(k-1)m^4 2^{\beta n}}{\epsilon^4}\right)$. ■