

# Extractors Using Hardness Amplification

Anindya De\* and Luca Trevisan\*\*

Computer Science Division  
University of California, Berkeley, CA, USA  
{anindya, luca}@cs.berkeley.edu

**Abstract.** Zimand [24] 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  $\gamma N$ , 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. [6], 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 [8], 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 [7] 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.

## 1 Introduction

Randomness extractors, defined by Nisan and Zuckerman [25,13] 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  $(K, \epsilon)$ -extractor if, for every random variable  $X$  of min-entropy at least  $K$ , the distribution  $Ext(X, U_t)$  has statistical distance at most  $\epsilon$  from the uniform distribution over  $\{0, 1\}^m$ .<sup>1</sup>

---

\* Supported by the "Berkeley Fellowship for Graduate Study".

\*\* This material is based upon work supported by the National Science Foundation under grant No. CCF-0729137 and by the US-Israel BSF grant 2006060.

<sup>1</sup> We 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  $\epsilon$  in statistical distance if for every statistical test  $T : \{0, 1\}^m \rightarrow \{0, 1\}$  we have

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

Besides their original applications to extract randomness from weak random sources and as primitives inside pseudorandom generators for space bounded computation, extractors have found several other applications. As surveyed in [12,16] 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 [10,1,9]. For such applications, 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 [2], however, is sufficient.)

The starting point of our paper is Zimand’s [24] simple construction of a locally computable extractor based on the Blum-Micali-Yao pseudorandom generator, and his analysis via the reconstruction approach of [20]. The extractor is neither optimal in terms of the output length nor the seed length. For e.g., both Lu [9] and Vadhan [21] achieve an optimal seed length of  $\Theta(\log n)$  for inverse polynomial error while extracting almost all the entropy of the source. In fact, [21] does better than [9] by extracting all but an arbitrarily small constant factor of the min-entropy while the latter has to lose an arbitrarily small polynomial factor. However, both these constructions are complicated in the sense that while Vadhan uses tools like samplers and extractors [15,26] from pseudorandomness machinery, Lu uses the extractor from [20] along with error-correcting codes based on expander graphs. In contrast, the extractor construction in Zimand [24] is extremely simple, only the analysis is non-trivial.

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  $(K, \epsilon)$  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 let  $\bar{p}$  be a permutation obtained from  $p$  via a *hardness amplification* procedure, so that the ability to invert  $\bar{p}$  on a small  $\alpha$  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  $BMY(\bar{p}, z)$ , the Blum-Micali-Yao generator applied to permutation  $\bar{p}$  with seed  $z$ . If 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  $\gamma N$ , 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 construction<sup>2</sup>, 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 [4]. Our analysis via Levin's XOR lemma is rather different from the one in [4] which is based on information-theoretic arguments. It should be noted that using information theoretic arguments, Dziembowski and Maurer manage to get an output length of  $N^{1-o(1)}$ . However, at a conceptual level, we show that the same style of analysis can be used both for the extractor in [4] and [24]<sup>3</sup>.

Using the *derandomized* direct product theorem of Impagliazzo and Wigderson [7], 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  $\epsilon$ , then there is a predictor  $P$ , derived from  $T$ , and  $i \leq m$  such that

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

Note that if the right-hand side of (1) were  $1 - \delta$  for some small  $\delta$ , 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  $H(\cdot)$  is the entropy function), and so we would be done.

<sup>2</sup> We 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 well for both the constructions but is slightly notationally cumbersome for the first one.

<sup>3</sup> The fact that [4] gets a better output length suggests that neither the original analysis of [24] nor our improved analysis is tight.

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

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

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

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

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

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

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

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

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

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  $\overline{x}$ , each of them containing  $x$  in a certain coordinate and fixed values everywhere else. For each such point,  $F(\overline{x} - \mathbf{1}), \dots, F(\overline{x} - \mathbf{i})$  can be specified using  $i \cdot (k - 1) \leq mk$  bits of advice. Overall, we need  $m^3k/\epsilon^2$  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

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

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

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

Then using appropriate versions of Goldreich-Levin and of the direct product lemma of Impagliazzo et al. [6], we can show that the construction is an

extractor provided that  $m$  is about  $N^{1/3}$  <sup>4</sup>. Construction (4) is precisely the second construction by Zimand [24].

By applying the *derandomized* direct product theorem of Impagliazzo and Wigderson [7], 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. Improved analysis of the construction by Zimand [24] as well as the description and proof of the *derandomized* extractor are deferred to the full version of the paper.

## 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 1.** [15,25] *Ext* :  $\{0, 1\}^N \times \{0, 1\}^t \rightarrow \{0, 1\}^m$  is said to be a  $(K, \epsilon)$  extractor if for every random variable  $X$  with min-entropy at least  $K$ , the statistical distance between output of the extractor and the uniform distribution is at most  $\epsilon$  i.e.  $\|Ext(X, U_t) - U_m\| \leq \epsilon$ . *Ext* is said to be a strong extractor if the seed can be included with the output and the distribution still remains close to uniform i.e.  $\|U_t \circ Ext(X, U_t) - U_{t+m}\| \leq \epsilon$ . 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  $\epsilon$  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 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

---

<sup>4</sup> Even using the ‘concatenation lemma’ of Goldreich et al. [5] which is a much more non-uniform version of the direct product theorem, we get  $m = N^{\frac{1}{10}}$  for which is better than Zimand’s analysis for entropy rates  $< 0.3$ .

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  $k$  be 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  $\gamma N$

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

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

**Observation 1.** *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*

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

In order to bound the number of functions which satisfy (6), we use the reconstruction approach in [20]<sup>5</sup> (and more generally used in the context of pseudorandom generators in [3,14]). 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

---

<sup>5</sup> This particular instance of reconstruction paradigm was used in context of extractors by Zimand [24] and earlier in context of pseudorandom generators by Blum, Micali and Yao [3,23].

$m > i \geq 0$ , using  $c$  bits of advice, we can construct  $C_f$  which satisfies (7) for some  $s \leq \frac{1}{2}$ .

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

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

**Lemma 1.** *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.** *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 1, 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 [20,19,17]. We do not redo the argument here but simply state the final result.

**Lemma 2.** *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).*

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

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 one of our constructions and its proof of correctness (Refer to the full version for the other two constructions).

### 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.  $Ext(f, \bar{y})_i = \bigoplus_{j=1}^k f(x_j + i - 1)$ . 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 [4].** As we have mentioned before, the construction in [4] is very similar though we have some minor simplifications. The extractor in [4]  $Ext' : (\{0, 1\}^{N+m-1})^k \times \{0, 1\}^{k \log N} \rightarrow \{0, 1\}^m$  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  $\bigoplus_{j=1}^k f_j(l_j + i - 1)$ . Thus, we avoid a minor complication of not having to divide the source into chunks. 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 [4].

#### Main Theorem and Proof of Correctness

**Theorem 3.** *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)  $\gamma > 0$ ,  $\epsilon \geq 2^{-\frac{m}{\gamma}}$ ,  $m = \frac{\epsilon^{\frac{2}{3}} 2^{\frac{m}{3}}}{n^2}$  and seed length  $kn = O\left(\frac{n \log \frac{m}{\epsilon}}{\gamma^2}\right)$ .*

Before proving Theorem 3, we see an immediate corollary of the above theorem with parameters of interest.

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

In order to prove Theorem 3, we first state the following main technical lemma of this section and then see how Theorem 3 follows from it. Subsequently, we prove the lemma.

**Lemma 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 using Lemma 3, it is useful to mention that an application of  $\delta$  is meaningful when it is close to 1 rather than 0. As can be seen from Lemma 3, we construct a circuit  $C_f$  which has correlation  $\delta$  with  $f$  and hence we would like  $1 - \delta$  to be small. This is different from the terminology used in Section 1 where we want to construct a circuit  $C_f$  which computes  $f$  with probability  $1 - \delta$  and hence we would like  $\delta$  to be close to 0.

*Proof (of Theorem 3).* In light of Observation 1, 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  $\delta$  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  $\gamma$  and sufficiently large  $n$ . With this, we satisfy the conditions for applying lemma 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 1, 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 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).

$$2^{\frac{6nk^2m^3}{\epsilon^2} + (\frac{1-\delta}{2} + H(\frac{1-\delta}{2}))2^n + m} \leq 2^{O(\frac{2^n}{n^3\gamma^4})} 2^{\frac{\gamma}{2}2^n} 2^{2^{\frac{n}{4}}} \leq 2^{-\frac{n}{8}} 2^{\gamma 2^n} \leq \epsilon 2^{\gamma 2^n}$$

*Proof ( of Lemma 3).* Using lemma 2, 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

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

In the above,  $x_1, x_2, \dots, x_k$  are independent random variables drawn from  $U_n$  and  $\bar{x}$  is the concatenation of  $x_1, \dots, x_k$ . 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

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

In this proof, we closely follow the proof of XOR lemma due to Levin [8] 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.

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

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  $\bar{f}(\bar{x} - s)$ . 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  $\delta$  and  $\eta$  be such that  $\delta^k \leq \frac{\epsilon}{m}$  and  $\eta = \frac{\epsilon}{km}$ . Then the above equation can be rewritten as

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

Further, we can write  $\bar{x}$  as  $x_1 \circ \bar{y}_1$  where  $x_1 \in \{0, 1\}^n$  and  $\bar{y}_1 \in (\{0, 1\}^n)^{k-1}$  and then the above can be rewritten as

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

where  $\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j)) = \mathbb{E}_{\bar{y}_1 \in U_{(k-1)n}}[\bar{f}(\bar{y}_1)C(x_1 \circ \bar{y}_1, \otimes_{j=1}^i f(x_1 - j)\bar{f}(\bar{y}_1 - j))]$ . At this stage, there are the following two possibilities.

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

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

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

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

*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  $\Gamma_1$ , then we could compute  $f(x_1)$  with high probability. Since computing  $\Gamma_1$  looks unlikely (without using  $2^n$  bits of advice), we will approximate  $\Gamma_1$  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  $\eta$  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  $\Gamma_2$  which is defined as follows. (Note

that  $\Gamma_2$  depends upon  $\overline{w}_i$ 's and the corresponding advice though  $\overline{w}_i$ 's are not explicitly included in the argument)

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

By Chernoff bound, we can say the following is true for all  $x_1$ . (The probability is over the random choices of  $\overline{w}_l$  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  $\Gamma_3$  as follows.

1. If  $|\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j))| \leq \delta^{k-1} + (k - 1)\eta$  then  $\Gamma_3$  is the same as  $\Gamma_2$  i.e.  $\Gamma_3(x_1, \otimes_{j=1}^i f(x_1 - j)) = \Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j))$
2. If not, then  $\Gamma_3$  has absolute value  $\delta^{k-1} + (k - 1)\eta$  with sign same as  $\Gamma_2$  i.e.

$$\Gamma_3(x_1, \otimes_{j=1}^i f(x_1 - j)) = \frac{|\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j))|}{(\Gamma_2(x_1, \otimes_{j=1}^i f(x_1 - j)))} (\delta^{k-1} + (k - 1)\eta)$$

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  $\Gamma_3$  is definitely at least as good a approximation of  $\Gamma$  as  $\Gamma_2$  is, we can say the following (the probability is again over the random choices of  $\overline{w}_l$  for  $l \in [q]$  and as before  $\overline{w}_l$  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  $\otimes_{l=1}^q \overline{w}_l$  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  $\otimes_{l=1}^q \overline{w}_l$ . Further, the advice required for getting  $\otimes_{j=0}^i \overline{f}(\overline{w}_l - j)$  for each  $l \in [q]$  is  $(i + 1)q \leq \frac{2n\eta m}{\eta^2}$  bits. So, we hardwire these 'good' values of  $\overline{w}_l$  and  $\otimes_{j=0}^i \overline{f}(\overline{w}_l - j)$  into  $C_1$  (i.e. instead of taking random choices, it now works with these hardwired values) and we can say that

$$\mathbb{E}_{x_1}[f(x_1)C_1(x_1, \otimes_{j=1}^i f(x_1 - j))] \geq \mathbb{E}_{x_1}[f(x_1)\Gamma(x_1, \otimes_{j=1}^i f(x_1 - j))] - \eta \tag{12}$$

The above claim uses that the range of  $f$  is  $[-1, 1]$ . This can now be combined with (10) to give the following

$$\mathbb{E}_{x_1}[f(x_1)C_1(x_1, \otimes_{j=1}^i f(x_1 - j))] > \delta^k + (k - 1)\eta \tag{13}$$

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 (13), we can say (using  $\delta \leq 1$ )

$$\mathbb{E}_{x_1}[f(x_1)C_2(x_1, \otimes_{j=1}^i f(x_1 - j))] > \frac{\delta^k + (k - 1)\eta}{\delta^{k-1} + (k - 1)\eta} \geq \delta \tag{14}$$

$C_2$  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)<sup>6</sup>. 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.

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

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

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

$$C^0(\overline{y_1}, \otimes_{j=1}^i \overline{f}(\overline{y_1} - j)) = (-1)^\alpha C(x_1 \circ \overline{y_1}, \otimes_{j=1}^i f(x_1 - j) \overline{f}(\overline{y_1} - j))$$

By definition and the previous assumptions, we get the following

$$\mathbb{E}_{\overline{y_1} \in U_{(k-1)n}} \overline{f}(\overline{y_1}) C^0(\overline{y_1}, \otimes_{j=1}^i \overline{f}(\overline{y_1} - j)) > \delta^{k-1} + (k-1)\eta$$

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 4, we require at most  $\frac{4nm}{\eta^2} + \log \left( \frac{4n}{\eta^2} \right) + 1$  bits of advice<sup>7</sup>. 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

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

<sup>6</sup> We 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.

<sup>7</sup> Note that  $\eta$  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 4 is that  $m \geq kn$  which shall definitely continue to hold as  $k$  decreases.

Finally accounting for the advice to use Lemma 2, 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 [22] 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.

**Acknowledgments.** The authors would like to thank Madhur Tulsiani for his useful comments on an earlier draft. The first author would like to also thank him for some very enjoyable discussions throughout the course of the work.

## References

1. Aumann, Y., Ding, Y.Z., Rabin, M.O.: Everlasting security in the bounded storage model. *IEEE Transactions on Information Theory* 48(6), 1668–1680 (2002)
2. Bar-Yossef, Z., Reingold, O., Shaltiel, R., Trevisan, L.: Streaming computation of combinatorial objects. In: *Proceedings of the 17th IEEE Conference on Computational Complexity*, pp. 165–174 (2002)
3. Blum, M., Micali, S.: How to generate cryptographically strong sequences of pseudorandom bits. *SIAM Journal on Computing* 13(4), 850–864 (1984); Preliminary version in *Proc. of FOCS 1982*
4. Dziembowski, S., Maurer, U.: Optimal randomizer efficiency in the bounded-storage model. *Journal of Cryptology* 17(1), 5–26 (2004); Preliminary version in *Proc. of STOC 2002*
5. Goldreich, O., Nisan, N., Wigderson, A.: On Yao’s XOR lemma. Technical Report TR95-50, *Electronic Colloquium on Computational Complexity* (1995)
6. Impagliazzo, R., Jaiswal, R., Kabanets, V., Wigderson, A.: Direct product theorems: Simplified, Optimized and Derandomized. In: *Proceedings of the 40th ACM Symposium on Theory of Computing*, pp. 579–588 (2008)
7. Impagliazzo, R., Wigderson, A.:  $P = BPP$  unless  $E$  has sub-exponential circuits. In: *Proceedings of the 29th ACM Symposium on Theory of Computing*, pp. 220–229 (1997)
8. Levin, L.: One-way functions and pseudorandom generators. *Combinatorica* 7(4), 357–363 (1987)

9. Lu, C.-J.: Encryption against storage-bounded adversaries from on-line strong extractors. *Journal of Cryptology* 17(1), 27–42 (2004)
10. Maurer, U.M.: Conditionally-perfect secrecy and a provably-secure randomized cipher. *J. Cryptology* 5(1), 53–66 (1992)
11. Nisan, N.: Extracting randomness: How and why. In: *Proceedings of the 11th IEEE Conference on Computational Complexity*, pp. 44–58 (1996)
12. Nisan, N., Ta-Shma, A.: Extracting randomness: A survey and new constructions. *Journal of Computer and System Sciences* (1998) (to appear); Preliminary versions in [11, 18]
13. Nisan, N., Zuckerman, D.: More deterministic simulation in Logspace. In: *Proceedings of the 25th ACM Symposium on Theory of Computing*, pp. 235–244 (1993)
14. Nisan, N., Wigderson, A.: Hardness vs randomness. *Journal of Computer and System Sciences* 49, 149–167 (1994); Preliminary version in *Proc. of FOCS 1988*
15. Nisan, N., Zuckerman, D.: Randomness is linear in space. *Journal of Computer and System Sciences* 52(1), 43–52 (1996); Preliminary version in *Proc. of STOC 1993*
16. Shaltiel, R.: Recent developments in extractors. *Bulletin of the European Association for Theoretical Computer Science* 77, 67–95 (2002)
17. Shaltiel, R., Umans, C.: Simple extractors for all min-entropies and a new pseudorandom generator. *Journal of the ACM* 52(2), 172–216 (2005)
18. Ta-Shma, A.: On extracting randomness from weak random sources. In: *Proceedings of the 28th ACM Symposium on Theory of Computing*, pp. 276–285 (1996)
19. Ta-Shma, A., Zuckerman, D., Safra, S.: Extractors from Reed-Muller codes. Technical Report TR01-036, *Electronic Colloquium on Computational Complexity* (2001)
20. Trevisan, L.: Extractors and pseudorandom generators. *Journal of the ACM* 48(4), 860–879 (2001)
21. Vadhan, S.P.: Constructing locally computable extractors and cryptosystems in the bounded-storage model. *Journal of Cryptology* 17(1), 43–77 (2004)
22. Viola, E.: The complexity of constructing pseudorandom generators from hard functions. *Computational Complexity* 13(3-4), 147–188 (2004)
23. Yao, A.C.: Theory and applications of trapdoor functions. In: *Proceedings of the 23th IEEE Symposium on Foundations of Computer Science*, pp. 80–91 (1982)
24. Zimand, M.: Simple Extractors via Constructions of Cryptographic Pseudo-random Generators. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) *ICALP 2005*. LNCS, vol. 3580, pp. 115–127. Springer, Heidelberg (2005)
25. Zuckerman, D.: General weak random sources. In: *Proceedings of the 31st IEEE Symposium on Foundations of Computer Science*, pp. 534–543 (1990)
26. Zuckerman, D.: Simulating BPP using a general weak random source. *Algorithmica* 16(4/5), 367–391 (1996)