COMBINATORICA
Bolyai Society – Springer-Verlag

COMBINATORICA **22** (1) (2002) 123–145

ALMOST OPTIMAL DISPERSERS

AMNON TA-SHMA*

Received November 12, 1998
Revised June 20, 2000

A (K,ϵ) disperser is a bipartite graph $G=(V_1,V_2,E)$ with the property that every subset A of V_1 of cardinality at least K, has at least $1-\epsilon$ fraction of the vertices of V_2 as neighbors. Such graphs have many applications in derandomization. Saks, Srinivasan and Zhou presented an explicit construction of a $(K=2^k,\epsilon)$ disperser $G=(V_1=[2^n],V_2,E)$ with an almost optimal degree $D=poly(\frac{n}{\epsilon})$, for every $k\geq n^{\Omega(1)}$. We extend their result for every parameter $k\geq poly\log(\frac{n}{\epsilon})$.

1. Introduction

A disperser is a graph with strong random-like properties. Formally,

Definition 1. (disperser) A bipartite graph $G=(V_1=[N=2^n],V_2=[M],E)$ is a (K,ϵ) disperser, if every subset $A\subseteq V$ of cardinality at least K has at least $(1-\epsilon)M$ distinct neighbors in V_2. If every vertex $v_1\in V_1$ has the same number D of neighbors, we call D the *degree* of the graph.

One such graph is, e.g., the complete bipartite graph and it has degree M. Sipser [10] noted that, in fact, very sparse dispersers exist and almost any degree $D=O(\frac{\log N}{\epsilon})$ bipartite graph is a (K,ϵ) disperser. Sipser left open the problem of explicitly constructing such graphs.

Mathematics Subject Classification (2000): 68Q01, 68R10

* This work was partially done while the author was at the Department of Computer Science, Weizmann Institute of Science, Rehovot, Israel. This work was supported by a Phil Zacharia postdoctoral fellowship.

Sipser also showed that the strong random-like properties of dispersers imply non-trivial derandomization results, and many other applications followed, e.g., for deterministic amplification [10,2], oblivious sampling [17], leader election [17], the hardness of approximating $\log\log \mathit{Clique}$ [16], explicit constructions of small depth super-concentrators, a-expanding graphs and non blocking networks [14], and more. The interested reader is referred to the survey paper [5] where most of these applications are discussed[1] . Yet, all these applications require an *explicit* construction of sparse dispersers:

Definition 2. (explicit disperser) A disperser $G=(V_1,V_2,E)$ of degree D is explicit if there is a polynomial time Turing machine that on input $v_1 \in V_1$ and $i \in [1 \dots D]$ computes the i'th neighbor of v_1.

We stress that the running time is polynomial in the input length, i.e., $\log N + \log D$.

Some of the applications require a small *entropy loss*, which we define next. Ideally, K vertices of a degree D graph can have KD neighbors. However, a lower bound of [9] shows that in any (K,ϵ) disperser $G=(V_1,V_2=[M],E)$ the size of V_2 must be smaller than KD. The entropy loss of a disperser is the log of this loss, i.e., $\log(\frac{KD}{M}) = \log(K) + \log(D) - \log(M)$. The [11] disperser has $n^{\Omega(1)}$ entropy loss, while the lower bound and the non-explicit constructions have only $\log\log\frac{1}{\epsilon} + O(1)$ entropy loss [9]. Reducing the entropy loss to the optimal is an important open problem with nice applications, e.g. for the construction of explicit a-expanding graphs and depth 2 super-concentrators.

Constructing explicit sparse dispersers turned out to be a non-trivial challenge. The problem was discussed, e.g., in [10,15,18,6,12,11,17] with incremental progress. The previous best construction was due to Saks, Srinivasan and Zhou [11] who showed an explicit disperser construction with degree $D = \mathit{poly}(\frac{n}{\epsilon})$ for sets of size $K=2^k$ with $k \ge n^{\Omega(1)}$. We show:

Theorem 1. *For every $\epsilon > 0$ and every $k \le n$, there is an explicit $(K=2^k,\epsilon)$ disperser $G=(V_1=[N=2^n], V_2=[M=2^m], E)$ with*

- *Degree $D \le \mathit{poly}(\frac{n}{\epsilon})$, and,*
- *Entropy loss $\mathit{poly}\log(\frac{n}{\epsilon})$.*

Thus our result improves on previous ones in two ways. First, the construction works for every $k \ge \mathit{poly}\log(\frac{n}{\epsilon})$ compared to $k \ge n^{\Omega(1)}$ in [11], and

[1] We note that [5] has a slightly different notation: a $(K=2^k,\epsilon)$ graph in our notation is called a (k,ϵ) graph in [5].

second, the construction has at most $poly\log(\frac{n}{\epsilon})$ entropy loss compared to $n^{\Omega(1)}$ in [11]. We summarize this in Table1.

Degree D	$k=\log K$	entropy loss	reference
$D=poly(\frac{n}{\epsilon})$	$k=n^{\Omega(1)}$	$n^{\Omega(1)}$	[11]
$D=poly(\frac{n}{\epsilon})$	$k\geq poly\log(\frac{n}{\epsilon})$	$poly\log(\frac{n}{\epsilon})$	This paper
$D=\Theta(\frac{n}{\epsilon})$	any k	$\log\log(\frac{1}{\epsilon})+O(1)$	Non-explicit [10,9]
$D=\Theta(\frac{n}{\epsilon})$	any k	$\log\log(\frac{1}{\epsilon})+O(1)$	Lower bound [6,9]

2. Preliminaries

2.1. Probability distributions and Random Variables

A probability distribution X over Λ, is a function $X:\Lambda\rightarrow[0,1]$ such that $\Sigma_{a\in\Lambda}X(a)=1$. We define the *distance* between two distributions using the l_1 norm:

Definition 3. (statistical distance) Two distributions X and Y over the same space Λ have statistical distance

$$d(X,Y)=\frac{1}{2}|X-Y|_1=\frac{1}{2}\sum_{a\in\Lambda}|X(a)-Y(a)|.$$

If $d(X,Y)\leq\epsilon$ we say X is ϵ close to Y.

We measure the amount of randomness in a distribution using min-entropy:

Definition 4. *(min-entropy)* The min-entropy of a distribution X is $H_\infty(X)=\min_a(-\log_2 X(a))$,

If X is a distribution then $x\in X$ denotes picking x according to the distribution X. U_n is the uniform distribution over $\{0,1\}^n$.

Let A and B be two possibly correlated random variables. We denote by $A\circ B$ the random variable that takes value (a,b) with probability $\Pr(A=a\wedge B=b)$ and by $A\times B$ the random variable that takes value (a,b) with probability $\Pr(A=a)\cdot\Pr(B=b)$.

With each random variable X we associate a probability distribution $\overline{A}:\Lambda\rightarrow[0,1]$ by letting $\overline{A}(a)=\Pr(A=a)$. We say two random variables A and B are ϵ close to each other, and we write $d(A,B)\leq\epsilon$, if the associated probability distributions are ϵ close to each other, i.e., $d(\overline{A},\overline{B})\leq\epsilon$. We define the min-entropy of a random variable A, $H_\infty(A)$, to be $H_\infty(\overline{A})$.

The following is a useful lemma from [6] that says that if X and Y are two correlated random variables such that for every $x \in X$ the conditional distribution $(Y \mid X = x)$ is close to the uniform distribution, then X and Y are (almost) uncorrelated. I.e., picking an element from the joint distribution of $X \circ Y$ is almost the same as picking $x \in X$ and *independently* $y \in Y$.

Lemma 1. [6] *Suppose X and Y be two correlated random variables such that for every $x \in X$ we have that $(Y|X = x)$ is ϵ close to uniform, then $X \circ Y$ is ϵ close to $X \times U$.*

2.2. Extractors

An extractor is a generalization of a disperser.

Definition 5. [6] (extractor for a class of distributions C) Let C be a set of distributions X over $\{0,1\}^n$. A function $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is an ϵ extractor for C, if for any distribution $X \in C$

$$d(E(X, U_d), U_m) \le \epsilon$$

where $E(X, U_d)$ is the distribution over $\{0,1\}^m$ obtained by choosing $x \in X$, and $y \in_U \{0,1\}^d$ and computing $E(x,y)$, and U_m is the uniform distribution over $\{0,1\}^m$.

We say E is explicit if $E(x,y)$ can be computed in time polynomial in the input length $|x| + |y| = n + d$.

Definition 6. A (k,ϵ) extractor is a function $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ that is an ϵ extractor for the set of all distributions X having at least k min-entropy.

There is a natural translation between extractors as defined in Definition 6 and graphs. Given a function $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ build a bipartite graph $G = (V_1 = [N = 2^n], V_2 = [M = 2^m], E)$ as follows. Identify V_1 with the set $\{0,1\}^n$ and V_2 with the set $\{0,1\}^m$. Include an edge (v_1, v_2) in E if and only if there is some $y \in \{0,1\}^d$ such that $E(v_1, y) = v_2$. If E is a (k,ϵ) extractor than the resulting graph is a $(K = 2^k, \epsilon)$ disperser, as can be seen by taking the uniform distribution over A in Definition 1 and invoking Definition 6. Informally we say that every extractor is a disperser. The converse is, however, false. For more information about extractors and their relationship to dispersers we refer the reader to [5].

2.3. Block-wise extractors

The ultimate goal is finding an optimal explicit extractor for all distributions having k min-entropy. However, it turns out that the problem is much simpler when we restrict ourselves to random sources having more structure:

Definition 7. [1] (block-wise source) Suppose X is a random variable taking values from $\{0,1\}^n$, and π is a partition of $[1 \ldots n]$ into l consecutive blocks. Define the induced random variable X_i^π to be the random variable X when restricted to the i'th block of π. Thus, $X = X_1^\pi \circ \ldots \circ X_l^\pi$ where the X_i^π are possibly correlated.

We say X is a $(\pi, z_1, \ldots, z_l)$ block-wise source, if for every $x \in \{0,1\}^n$ for which $\Pr(X = x) > 0$ and for every $1 \le i \le l$ we have $H_\infty(X_i^\pi \mid X_{i-1}^\pi = x_{i-1}, \ldots, X_1^\pi = x_1) \ge z_i$. Many times we omit the partition π and simply say that X is a $(z_1, \ldots, z_l)$ block-wise source.

Example 1.

- Consider the random variable that takes value $x \in \{0,1\}^n$ with probability 2^{-n}, for all $x \in \{0,1\}^n$. Clearly any partition point partitions the source into two blocks each with min-entropy that equals its block length.
- Now consider the random variable A on $\{0,1\}^{2n+1}$ that takes values $0^n x 0$ and $x 0^n 1$ with probability 2^{-n+1}, for all $x \in \{0,1\}^n$. There is no partition π under which X is a (π, z_1, z_2) block-wise source with $z_1, z_2 > 1$, for if we partition $[1 \ldots 2n+1]$ at some point $i \in [1 \ldots n]$ then with probability $1/2$ the first block is the 0 string, while if we put the partition point i in the second half then for a non-zero prefix the second block is fixed.

We say a distribution X on $\{0,1\}^n$ is a $(\pi, z_1, \ldots, z_l)$ block-wise source, if the random variable that takes value $a \in \{0,1\}^n$ with probability $X(a)$ is a $(\pi, z_1, \ldots, z_l)$ block-wise source. A block-wise extractor is an extractor that works for all block-wise sources:

Definition 8. (a block-wise extractor) A $((z_1, \ldots, z_l), \epsilon)$ block-wise extractor is a function $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ that is an ϵ extractor for the set of all $(\pi; z_1, \ldots, z_l)$ block-wise sources X. E may depend on π (and $z_1, \ldots, z_l$ and ϵ), and when we want to emphasize this we write E_π.

Good block-wise extractors were constructed in [6, 12] and we will discuss them later.

2.4. Somewhere random extractors

Next we define an object that will turn out to be stronger than a disperser but weaker than an extractor. We start with a definition of a somewhere random source:

Definition 9. [5] (somewhere random source) $B=(B_1,\dots,B_b)$ is a b–block (m,ϵ,η) somewhere random source if each B_i is a random variable over $\{0,1\}^m$ and there is a random variable Y over $[0\dots b]$ such that:

- For every $i \in [1\dots b]$: $\Pr(Y=i)>0 \implies d((B_i|Y=i),U_m)\leq \epsilon$.
- $\Pr(Y=0)\leq\eta$.

We call Y an (ϵ,η) selector for B.

We think of a somewhere random source as a bunch of correlated random variables that are also correlated with some (possibly unknown) selector function that tells which of them is the uniform one. The case $Y=0$ means the selector function could not find an appropriate block and this happens with probability at most η.

Example 2. Consider again the random variable A of example 1. Let A_1 be the random variable that outputs the first n bits of A (0^n with probability half, and uniform otherwise) and A_2 the random variable that outputs the last n bits (again, 0^n with probability half, and uniform otherwise). Clearly, A_1 and A_2 are correlated. Now we choose the selector function to be 1 if the last bit is 1 and 2 otherwise. This defines a two-block $(n,0,0)$ somewhere random source.

We say a distribution X on $(\{0,1\}^m)^b$ is an (m,ϵ,η) somewhere random source, if the random variable that takes value $x_1,\dots,x_b \in \{0,1\}^{mb}$ with probability $X(x_1,\dots,x_b)$ is an (m,ϵ,η) somewhere random source. We define a somewhere random extractor to be a function whose output is a somewhere random source.

Definition 10. (somewhere random extractor) Let $S:\{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^b$ be a function. Given a distribution X on $\{0,1\}^n$ the distribution $S(X,U_d)=B_1\circ\dots\circ B_b$ is obtained by picking $x\in X, y\in U_d$ and computing $S(x,y)$.

We say S is a (k,ϵ,η) *somewhere random extractor* if for every distribution X with $H_\infty(X)\geq k$, $\{B_1,\dots,B_b\}$ is a b-block (m,ϵ,η) somewhere random source. When $\epsilon=\eta$ we say S is a (k,ϵ) somewhere random extractor.

Given a random source with k min-entropy, a (k,ϵ) extractor outputs a single distribution that is ϵ close to uniform. In contrast, a somewhere random extractor may output many distributions with the guarantee that at least one of them (and possibly only one) is ϵ close to uniform. Thus, a somewhere random extractor is weaker than an extractor.

3. Top-level view of the proof

Theorem 1 asserts the existence of low degree dispersers. We prove that by showing the existence of explicit somewhere random extractors:

Theorem 2. *For every $\epsilon > 0$ and k, there is a (k,ϵ) somewhere random extractor $S : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^{n^2}$ with $d = O(\log(\frac{n}{\epsilon}))$ and $m = k - poly\log(\frac{n}{\epsilon})$.*

We will prove Theorem 1 in the next subsection. We then show that such somewhere random extractors give rise to good dispersers:

Lemma 2. *Let $\eta < 1$. Let $S : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^b$ be a (k,ϵ,η) somewhere random extractor. Suppose $S(x,y) = S_1(x,y) \circ \ldots \circ S_b(x,y)$. Define the bipartite graph $G = (V_1 = \{0,1\}^n, V_2 = \{0,1\}^m, E)$ where $(v_1,v_2) \in E$ iff there is some $1 \le i \le b$ and some $z \in \{0,1\}^d$ such that $v_2 = S_i(v_1,z)$. Then G is a $(K = 2^k, \epsilon)$ disperser.*

Proof. Let $X \subseteq V_1$ be of cardinality at least 2^k. Let $\overline{X}$ be the uniform distribution over X. Clearly, $H_\infty(\overline{X}) \ge k$. Since S is a (k,ϵ,η) somewhere random extractor, $S(\overline{X}, U_d)$ is an (m,ϵ,η) somewhere random source. Let $Y = Y(x,z)$ be a selector function. Since $\eta < 1$ there must be some $i > 0$ such that $\Pr(Y = i) > 0$ and $d((S_i(\overline{X}, U_d) \mid Y = i), U_m) \le \epsilon$. Thus, even when we restrict ourselves only to $x \in X$ and $z \in \{0,1\}^d$ such that $Y(x,z) = i$, these edges induce a distribution that is ϵ close to the uniform distribution over V_2. Thus, these edges miss at most an ϵ fraction of V_2. In particular, the set of all neighbors of X, $\Gamma(X)$ miss at most an ϵ fraction of V_2. ∎

Together, Theorem 2 and Lemma 2 give Theorem 1.

3.1. Building the somewhere random extractor

We now turn into building the somewhere random extractor of Theorem 2. Our goal is to build a somewhere random extractor that uses only $d = O(\log(\frac{n}{\epsilon}))$ truly random bits, and has a small entropy loss. On the other hand, if we relax our requirements and allow using $d = poly\log(n)\log\frac{1}{\epsilon}$ truly random bits, then such a somewhere random extractor, in fact even an extractor, is known and was constructed in [5,7]. Thus, if we could only bridge the gap between the $O(\log(\frac{n}{\epsilon}))$ truly random bits that we are allowed to use, to the $poly\log(n)\log\frac{1}{\epsilon}$ truly random bits that we would have liked to use, we could have our construction.

While it is not known how to do that in general, [6,12] show how to achieve that for the special case of a block-wise source. The main property that is used is the following. Say $X = X_1 \circ X_2$ is a block-wise source, and $y_1 = E(X_2, y_2)$ is the output of an extractor operating on the second block with a truly random string y_2, then y_1 can be used as the truly random string for further extracting randomness from X_1. Thus, we can start with $t_2 = |y_2|$ truly random bits, extract some additional randomness from the second block X_2, and get $t_1 = |y_1| > t_2$ (almost) truly random bits that can be used for further extracting some more randomness from X_1.

Yet, we would like to deal with more than just block-wise sources. Nisan and Zuckerman, and later Srinivasan and Zuckerman, tried to get general extractors by reducing a general source to a block-wise source, and then using the block-wise extractor. Saks, Srinivasan and Zhou took a different direction. They basically claimed that in every source $X = X_1 \circ \ldots \circ X_n$ there is a hidden block-wise source. The actual implementation of their idea is, in fact, quite complicated. We continue their work, simplify and strengthen it. Doing it, we get the following general composition procedure:

<u>$E \ominus S$</u>

Basic components:
- $S : \{0,1\}^n \times \{0,1\}^{d_1} \to (\{0,1\}^{d_2})^l$ a (k_1, ζ_1, η_1) somewhere random extractor,
- $E : \{0,1\}^n \times \{0,1\}^{d_2} \to \{0,1\}^{d_3}$ a (k_2, ζ_2)–extractor

Define $E \ominus S : \{0,1\}^n \times \{0,1\}^{d_1} \to (\{0,1\}^{d_3})^{nl}$ as follows:

Input: $a \in \{0,1\}^n$, $r_1 \in \{0,1\}^{d_1}$.
Output: For $1 \le i \le n$ and $1 \le j \le l$,
- Let q_i^j be the j'th block in the output of $S(a_{[i,n]} \circ 0^{i-1}, r_1)$. And,
- Let $z_i^j = E(a_{[1,i-1]} \circ 0^{n-i+1}, q_i^j)$.

The output contains nl blocks, where the (i,j)th output block is z_i^j.

From now on we write $S(a_{[i,n]}, r_1)$ instead of $S(a_{[i,n]} \circ 0^{i-1}, r_1)$, with the understanding that whenever the first argument of S has length smaller than

n, it is padded with additional zeros to become a length n string. A similar convention holds for E. Now, we claim:

Theorem 3. (Composition lemma) *Suppose S, E are as above. For every $s > 0$, $E \ominus S$ is a $(k_1 + k_2 + s, \zeta_1 + \zeta_2, \eta_1 + O(nl2^{-s/3}))$ somewhere random extractor.*

A similar theorem appears in [5] where a composition of two extractors is discussed. In our case we compose an extractor with a somewhere random extractor. The ideas are the same, but the implementation is a bit different. We prove Theorem 3 in Section 4.

3.2. Proof of Theorem 2

The somewhere random extractor we build is $E \ominus S$ where E is an extractor that extracts all the entropy of the source (i.e., it has optimal entropy loss) but uses $poly\log(n)\log\frac{1}{\epsilon}$ truly random bits instead of the optimal $O(\log(\frac{n}{\epsilon}))$:

Theorem 4. [7] *For every $k_2 \le n$ and $\epsilon > 0$ there exists an explicit (k_2, ϵ) extractor $E : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^{k_2+t-\Delta}$ with*

- *$d = O(\log^3 n \cdot \log\frac{1}{\epsilon})$, and*
- *$\Delta = 2\log\frac{1}{\epsilon} + O(1)$.*

The somewhere random extractor S uses optimal seed length $d = O(\log(\frac{n}{\epsilon}))$ and extracts $\Omega(\log^3 n \log\frac{1}{\epsilon})$ output bits:

Theorem 5. *For every $\epsilon > 0$ there is a $(k_1 = poly\log(\frac{n}{\epsilon}), \epsilon)$ somewhere random extractor $S : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^{n^2}$ with $m = \Theta(\log^3 n \log\frac{1}{\epsilon})$ and $d = O(\log(\frac{n}{\epsilon}))$.*

We prove Theorem 5 in Section 6. Let us now derive Theorem 2:

Proof. Consider $E \ominus S : \{0,1\}^n \times \{0,1\}^{d_1} \to (\{0,1\}^{d_3})^{n^3}$ where S and E are as above. We apply Theorem 3 with $\zeta_1 = \zeta_2 = \eta_1 = \Omega(\epsilon)$ and $s = O(\log(\frac{n}{\epsilon}))$. We also choose $k_2 = k - k_1 - s$. We see that $E \ominus S$ is a (k, ϵ) somewhere random extractor. The number of truly random bits used is $d_1 = O(\log(\frac{n}{\epsilon}))$ and the somewhere random extractor has n^3 blocks each of length $d_3 = k_2 + d_1 - \Delta - O(1) \ge k_2 = k - poly\log(\frac{n}{\epsilon})$. ∎

4. A proof of the composition lemma

We now turn to the proof of Theorem 3. We need to show that $E \ominus S$ is a (k,ϵ) somewhere random extractor, for the specified k and ϵ. I.e., we need to show that for every distribution X on $\{0,1\}^n$ with $H_\infty(X) \geq k$, the random variable $Z = E \ominus S(X,U)$ obtained by picking x according to the distribution X, y uniformly from $\{0,1\}^{d_1}$, and computing $E \ominus S(x,y)$ is a somewhere random source.

We adopt the following abbreviation: for $x \in \{0,1\}^n$ and $1 \leq a \leq b \leq n$ we let $x_{[a,b]}$ stand for the string $x_a \circ x_{a+1} \circ \ldots \circ x_b$. Similarly the random variable $X_{[a,b]}$ stands for $X_a \circ \ldots \circ X_b$.

Proof of Theorem 3. Let X be a distribution with $H_\infty(X) \geq k_1 + k_2 + s$. Denote by Q_i^j and Z_i^j $(i = 1,\ldots,n; j = 1,\ldots,l)$ the random variables with values q_i^j and z_i^j respectively. Also, let $\epsilon_3 = 2^{-s/3}$, $\epsilon_2 = 2\epsilon_3$, and $\epsilon_1 = \epsilon_3$. Define $Z^j = Z_1^j \circ \ldots \circ Z_n^j$ and $Z = Z^1 \circ \ldots \circ Z^l$. Thus, Z contains all the nl output blocks of $E \ominus S$. We need to show that Z is a somewhere random source. We start by defining a selector function for Z.

4.1. The selector function

We first give an informal explanation on how we build the selector function. Given $w \in \{0,1\}^n$ the selector function Y_1 selects $i \in [1 \ldots n]$ such that $\Pr(X_{[i,n]} = w_{[i,n]} \mid X_{[1,i-1]} = w_{[1,i-1]})$ is small. We take this i to be a partition point of the n bit string into two blocks, one with the first $i-1$ bits of w, and the second with the rest of the bits. We then look at $S(a_{[i,n]}, r_1)$ which contains l output blocks $q_i^1, \ldots, q_i^l$. S is a somewhere random extractor, so under the right conditions, there is a selector function for the random variables $Q_i^1, \ldots, Q_i^l$. Our selector function Y is one that first selects $1 \leq i \leq n$ according to Y_1, and then selects $1 \leq j \leq l$ according to the selector function of S. We now give the details:

Definition 11. Given $w \in \{0,1\}^n$ let $f_1(w)$ be the last i s.t $\Pr(X_{[i,n]} = w_{[i,n]} \mid X_{[1,i-1]} = w_{[1,i-1]}) \leq (\epsilon_2 - \epsilon_3) \cdot 2^{-k_1}$.

The following "bad" cases are defined below:

Definition 12. Define $w \in \{0,1\}^n$ to be "bad" if $f_1(w) = i$ and:

- $\Pr_{x \in X}(f_1(x) = i \mid x_{[1,i-1]} = w_{[1,i-1]}) \leq \epsilon_2$, or,
- $\Pr_{x \in X}(X_i = w_i \mid x_{[1,i-1]} = w_{[1,i-1]}) \leq \epsilon_3$

We denote by B the set of all bad w. If the first event happens we say $w \in B_2$ and if the second then $w \in B_3$.

We define our selector function to be f_1 whenever we do not have a bad w. I.e.,

Definition 13. Let Y_1 be the random variable obtained by taking the input a and letting $Y_1 = Y_1(a)$, where:

$$Y_1(w) = \begin{cases} 0 & w \text{ is bad} \\ f_1(w) & \text{otherwise} \end{cases}$$

We next observe that $H_\infty(X_{[i,n]} \mid Y_1 = i \text{ and } X_{[1,i-1]} = w_{[1,i-1]}) \geq k_1$, i.e., if we pick an input $w \in \{0,1\}^n$ according to the distribution $(X \mid Y_1 = i)$ and we partition the n bits into two blocks at location i, then the second block contains a lot of entropy even given the first block.

Claim 1. *If $i \neq 0$ and $\Pr(Y_1 = i \mid X_{[1,i-1]} = w_{[1,i-1]}) > 0$ then $H_\infty(X_{[i,n]} \mid Y_1 = i \text{ and } X_{[1,i-1]} = w_{[1,i-1]}) \geq k_1$.*

Proof of claim 1. For any w such that $Y_1(w) = i \neq 0$:

$$\begin{aligned} \Pr(X_{[i,n]} = w_{[i,n]} | X_{[1,i-1]} = w_{[1,i-1]}, Y_1(x) = i) &\leq \\ \frac{\Pr(X_{[i,n]} = w_{[i,n]} \mid X_{[1,i-1]} = w_{[1,i-1]})}{\Pr(Y_1(x) = i \mid X_{[1,i-1]} = w_{[1,i-1]})} &\leq \\ \frac{(\epsilon_2 - \epsilon_3) \cdot 2^{-k_1}}{\Pr(Y_1(x) = i \mid X_{[1,i-1]} = w_{[1,i-1]})} &\leq \\ \frac{(\epsilon_2 - \epsilon_3) \cdot 2^{-k_1}}{\epsilon_2 - \epsilon_3} &= 2^{-k_1} \end{aligned}$$

The first line holds since $\Pr(A \mid B) \leq \frac{\Pr(A)}{\Pr(B)}$, the second line since $f_1(w) = i$, and the third follows from Claim 5 whose proof appears in Section 4.4. ∎

Fix any w with $Y_1(w) = i \neq 0$. Denote

$$X_{w_{[1,i-1]},i} = (X_{[i,n]} \mid Y_1 = i \text{ and } X_{[1,i-1]} = w_{[1,i-1]}).$$

By the previous claim we know that $H_\infty(X_{w_{[1,i-1]},i}) \geq k_1$. Now S is a (k_1, ζ_1, η_1) somewhere random extractor, hence $S(X_{w_{[1,i-1]},i}, U_{d_1})$ is a (d_2, ζ_1, η_1) somewhere random source. Let $Y_{w_{[1,i-1]},i}$ be a selector function for it. We define:

Definition 14. Given w with $Y_1(w)=i\neq 0$ define $f_2(w)=Y_{w_{[1,i-1]},i}(w)\in[0,\dots,l]$.

Again, we get rid of bad values:

Definition 15. $w\in B_1$ if $Y_1(w)=i\neq 0$ and $f_2(w)=j\neq 0$ and $\Pr(Y_1=i \text{ and } f_2=j)\leq\epsilon_1$.

Finally, we define the selector function Y:

Definition 16.

$$Y(w)=\begin{cases}(0,0) & Y_1(w)=0 \text{ or } f_2(w)=0 \text{ or } w\in B_1\\ (Y_1(w),Y_2(w)) & \text{otherwise}\end{cases}$$

4.2. Y is a selector for Z

We now check that Y is a good selector for the source Z. I.e., we need to show that $\Pr(Y=(0,0))\leq\eta_1+4nl2^{-s/3}$ and that for $Y=(i,j)\neq(0,0)$ we have $(Z_i^j\,|\,Y=(i,j))$ is $\zeta_1+\zeta_2$ close to uniform.

- $\Pr(Y=(0,0))\leq\eta_1+4nl2^{-s/3}$: Clearly, $\Pr(Y=(0,0))\leq\Pr(x\in B_1)+\Pr(x\in B_2)+\Pr(x\in B_3)+\Pr(f_2=0\,|\,Y_1\neq 0)$. In Section 4.4 we give the (easy) proof that $\Pr(x\in B_i)\leq nl\epsilon_i$ for $i=1,2,3$. Also,

$$\begin{aligned}\Pr(f_2=0\mid Y_1\neq 0)&=\sum_i\sum_{w_{[1,i-1]}}\Pr(Y_1=i \text{ and } X_{[1,i-1]}=w_{[1,i-1]})\\&\qquad\cdot\Pr(f_2=0\mid Y_1=i \text{ and } X_{[1,i-1]}=w_{[1,i-1]})\\&\leq\sum_i\sum_{w_{[1,i-1]}}\Pr(Y_1=i \text{ and } X_{[1,i-1]}=w_{[1,i-1]})\cdot\eta_1\\&\leq\eta_1\end{aligned}$$

  The inequality holds because $H_\infty(X_{w_{[1,i-1]},i})\geq k_1$ and S is a (k_1,ζ_1,η_1) somewhere random extractor.
- $(Z_i^j\,|\,Y=(i,j))$ is $\zeta_1+\zeta_2$ close to uniform: Fix some $(i,j)\neq(0,0)$ with $\Pr(Y=(i,j))>0$ and any $w_{[1,i-1]}$ that can be extended to some w with $Y(w)=(i,j)$.

  Claim 2. *$(Q_i^j\mid Y_1=i$ and $X_{[1,i-1]}=w_{[1,i-1]}$ and $f_2=j)$ is ζ_1 close to uniform.*

Proof. The random variable $S(X_{w_{[1,i-1]},i})$ is composed of l correlated random variables $(Q_i^1|Y_1 = i \text{ and } X_{[1,i-1]=w_{[1,i-1]}}) \circ \ldots \circ (Q_i^l|Y_1 = i \text{ and } X_{[1,i-1]=w_{[1,i-1]}})$ that correspond to the l output blocks of the somewhere random extractor S. Now, we have seen before that $S(X_{w_{[1,i-1]},i}, U_{d_1})$ is a somewhere random source with the selector function $Y_{w_{[1,i-1]},i}$. In particular, this means that $(Q_i^j|Y_1 = i \text{ and } X_{[1,i-1]=w_{[1,i-1]}} \text{ and } Y_{w_{[1,i-1]},i}=j)$ is ζ_1 close to uniform. ∎

Now for any (i,j) with $\Pr(Y=(i,j))>0$ we have $Y=(i,j) \iff Y_1 = i$ and $f_2=j$. Hence:

Claim 3. *For any $w_{[1,i-1]}$ that can be extended to some w with $Y(w)=(i,j)$ we have $(Q_i^j \,|\, Y=(i,j)$ and $X_{[1,i-1]}=w_{[1,i-1]})$ is ζ_1 close to uniform.*

By Lemma 1 we conclude that the distribution $(X_{[1,i-1]} \circ Q_i^j \,|\, Y=(i,j))$ is ζ_1 close to the distribution $(X_{[1,i-1]} \,|\, Y=(i,j)) \times U$, where U is the uniform distribution over $\{0,1\}^{d_2}$.
Finally, we claim that the first block also contains a lot of entropy, namely,

Claim 4. *If $(i,j) \neq (0,0)$ and $\Pr(Y=(i,j))>0$ then $H_\infty(X_{[1,i-1]} \,|\, Y=(i,j)) \geq k_2$.*

we prove this soon. Having that, using the (k_2,ζ_2) extractor E we get that $(Z_i^j \,|\, Y=(i,j))$ is $\zeta_1+\zeta_2$ close to uniform, as desired.

∎

4.3. The first block has a lot of entropy

Proof of claim 4. Take any $w_{[1,i-1]}$ that can be extended to some w with $Y(w)=(i,j)\neq 0$.

$$\Pr(X_{[1,i-1]} = w_{[1,i-1]}) = \frac{\Pr(X_{[1,n]} = w_{[1,n]})}{\Pr(X_{[i,n]} = w_{[i,n]} \mid X_{[1,i-1]} = w_{[1,i-1]})} = \frac{\Pr(X_{[1,n]} = w_{[1,n]})}{\Pr(X_i = w_i|X_{[1,i-1]})\Pr(X_{[i+1,n]} = w_{[i+1,n]}|X_{[1,i]})}$$

However,

$$\Pr(X_{[i+1,n]} = w_{[i+1,n]}|X_{[1,i]}) \geq (\epsilon_2 - \epsilon_3)2^{-k_1}$$
$$\Pr(X_i = w_i \mid X_{[1,i-1]} = w_{[1,i-1]}) \geq \epsilon_3$$
$$\Pr(X_{[1,n]} = w_{[1,n]}) \leq 2^{-(k_1+k_2+s)}$$

The first line is true because $f_1(w)=i$, the second because $w\notin B_3$, and the third because $H_\infty(X)\geq k_1+k_2+s$. Thus,

$$\Pr(X_{[1,i-1]} = w_{[1,i-1]}) \leq \frac{2^{-k_2-s}}{\epsilon_3\cdot(\epsilon_2-\epsilon_3)} \tag{1}$$

Therefore,

$$\begin{aligned}
&\Pr(X_{[1,i-1]} = w_{[1,i-1]} \mid Y(x)=(i,j)) \leq \\
&\frac{\Pr(X_{[1,i-1]} = w_{[1,i-1]})}{\Pr(Y(x)=(i,j))} \leq \\
&\frac{2^{-k_2-s}}{\epsilon_3\cdot(\epsilon_2-\epsilon_3)\cdot\Pr(Y(x)=(i,j))} \leq \\
&\frac{2^{-k_2-s}}{\epsilon_3\cdot(\epsilon_2-\epsilon_3)\cdot\epsilon_1} = \frac{2^{-k_2-s}}{\epsilon_3^3} = \frac{2^{-k_2-s}}{2^{-s}} = 2^{-k_2}
\end{aligned}$$

The first line is true because $\Pr(A\mid B)\leq\frac{\Pr(A)}{\Pr(B)}$. The second follows from Eq. (1). The third follows because $Y(w)=(i,j)\neq(0,0)$ implies $w\notin B_1$ and hence $\Pr(Y=(i,j))\geq\epsilon_1$. ∎

4.4. Proof of the more technical lemmas

We now give the proofs of the more technical lemmas that were used above.

Claim 5. *For any i and any $w_{[1,i-1]}$, if $\Pr_{x\in X}(Y_1(x)=i\mid X_{[1,i-1]}=w_{[1,i-1]})>0$, then $\Pr_{x\in X}(Y_1(x)=i\mid X_{[1,i-1]}=w_{[1,i-1]})\geq\epsilon_2-\epsilon_3$.*

Proof. Since $w_{[1,i-1]}$ can be extended to some w with $Y_1(w)=i\neq 0$, by definition 12 $\Pr(f_1(x)=i\mid X_{[1,i-1]}=w_{[1,i-1]})\geq\epsilon_2$. This implies that for *any* extension w' of $w_{[1,i-1]}$ with $f_1(w')=i$, it holds that $w'\notin B_2$. Hence,

$$\begin{aligned}
&\Pr(Y_1(x)=i \mid X_{[1,i-1]}=w_{[1,i-1]}) = \\
&\Pr(f_1(x)=i \mid X_{[1,i-1]}=w_{[1,i-1]}) \\
&\qquad -\Pr(f_1(x)=i \text{ and } x\in B \mid X_{[1,i-1]}=w_{[1,i-1]}) = \\
&\Pr(f_1(x)=i \mid X_{[1,i-1]}=w_{[1,i-1]}) \\
&\qquad -\Pr(f_1(x)=i \text{ and } x\in B_3 \mid X_{[1,i-1]}=w_{[1,i-1]}) \geq \\
&\epsilon_2-\epsilon_3
\end{aligned}$$

The last inequality uses claim 6. ∎

Claim 6.

1. *For any i and $w_{[1,i-1]}$:* $\Pr(f_1(x)=i$ *and* $x\in B_3 \,|\, X_{[1,i-1]}=w_{[1,i-1]})\leq\epsilon_3$
2. *For any i:* $\Pr(f_1(x)=i$ *and* $x\in B_3)\leq\epsilon_3$
3. $\Pr(x\in B_3)\leq n\epsilon_3$

Proof.

1) If for some $w_{[1,i-1]}$ $\Pr(f_1(x)=i$ and $x\in B_3 \,|\, X_{[1,i-1]}=w_{[1,i-1]})>0$ then fix z to be the (unique) bit such that $\Pr(X_i=z \,|\, X_{[1,i-1]}=w_{[1,i-1]})\leq\epsilon_3$. Then $\Pr(f_1(x)=i$ and $x\in B_3 \,|\, X_{[1,i-1]}=w_{[1,i-1]})\leq\Pr(X_i=z \,|\, X_{[1,i-1]}=w_{[1,i-1]})\leq\epsilon_3$.
2) $\Pr(f_1(x)=i$ and $x\in B_3)\leq\sum_{w_{[1,i-1]}}\Pr(X_{[1,i-1]}=w_{[1,i-1]})\cdot\Pr(f_1(x)=i$ and $x\in B_3 \,|\, X_{[1,i-1]}=w_{[1,i-1]})\leq\sum_{w_{[1,i-1]}}\Pr(X_{[1,i-1]}=w_{[1,i-1]})\cdot\epsilon_3\leq\epsilon_3$.
3) $\Pr(x\in B_3)\leq\sum_{i=1}^{n}\Pr(x\in B_3$ and $f_1(x)=i)\leq n\epsilon_3$.

■

Claim 7.

1. *For any i:* $\Pr(f_1(x)=i$ *and* $x\in B_2)\leq\epsilon_2$
2. $\Pr(x\in B_2)\leq n\epsilon_2$.

Proof.

1) If for some $w_{[1,i-1]}$ $\Pr(f_1(x)=i$ and $x\in B_2 \,|\, X_{[1,i-1]}=w_{[1,i-1]})>0$ then there is an extension w of $w_{[1,i-1]}$ such that: $f_1(w)=i$ and $w\in B_2$, and therefore, $\Pr(f_1(x)=i \mid X_{[1,i-1]}=w_{[1,i-1]})\leq\epsilon_2$. Thus, for all $w_{[1,i-1]}$, $\Pr(f_1(x)=i$ and $x\in B_2 \mid X_{[1,i-1]}=w_{[1,i-1]})\leq\epsilon_2$. Therefore, $\Pr(f_1(x)=i$ and $x\in B_2)=\sum_{w_{[1,i-1]}}\Pr(x_{[1,i-1]}=w_{[1,i-1]})\cdot\Pr(f_1(x)=i$ and $x\in B_2 \,|\, X_{[1,i-1]}=w_{[1,i-1]})\leq\sum_{w_{[1,i-1]}}\Pr(X_{[1,i-1]}=w_{[1,i-1]})\cdot\epsilon_2\leq\epsilon_2$.
2) $\Pr(x\in B_2)\leq\sum_{i=1}^{n}\Pr(x\in B_2$ and $f_1(x)=i)\leq n\epsilon_2$. ■

Using similar arguments it is easy to see that $\Pr(x\in B_1)\leq nl\epsilon_1$.

5. A small family of segmentations

In this section we strengthen a combinatorial lemma appearing in [11]. Loosely speaking, the lemma claims that there is a small family of segmentations of $[1\ldots n]$ into few blocks such that for any possible way of dividing weight among the n elements in $[1\ldots n]$ there is at least one segmentation in the family such that every block in the segmentation has high weight.

Definition 17. (segmentations) We say π partitions $[1\dots n]$ into l blocks if it produces l segments $B_1=[1\dots b_1], B_2=[b_1+1\dots b_2], \dots, B_l=[b_{l-1}+1\dots n]$. We call π a segmentation.

A family F of segmentations of $[1\dots n]$ into l blocks is $(k,[z_1,\dots,z_l])$ good for any weight function $p:[1\dots n]\to[0,w]$ if for any such p with $\sum_i p(i)\geq k$ there is at least one segmentation $\pi\in F$ that partitions $[1\dots n]$ into blocks $B_1,\dots,B_l$ such that for every $1\leq i\leq l$ $\sum_{j\in B_i} p(j)\geq z_i$.

The next lemma is a generalization of a lemma appearing in [11]. The proof presents a somewhat simpler algorithm with a simpler analysis and better parameters. The idea behind the construction is, though, essentially the same as in [11].

Lemma 3. *Suppose $\frac{k}{2^l}-\sum_{i=1}^{l} z_i\geq w$ for some positive values k,l,n,z_i,w. Then there is a family F of segmentations of $[1\dots n]$ into l blocks that is $(k,[z_1,\dots,z_l])$ good for any $p:[1\dots n]\to[0,w]$, and such that the size of F is at most n^2.*

Proof. W.l.o.g. we can assume n is a power of two, for otherwise we can just add dummy zero weights. We take a balanced binary tree over n leaves. The leaves of the subtree whose root is v cover a consecutive subset of $[1\dots n]$ which we denote by $dom(v)$.

Construction of F:

- Take all possible paths from the root to a leaf.
- For each path $v_1,\dots,v_{\log n}$ take all subsets of size $l-1$ of $\{v_1,\dots,v_{\log n}\}$.
- Each such subset $\{v_{i_1},\dots,v_{i_{l-1}}\}$ of $l-1$ vertices determines a partitioning of $[1\dots n]$ into l blocks as follows: each v_{i_k} puts a partition point on the middle point of $dom(v_{i_k})$.

Clearly there are n possible paths, and each path can have at most $2^{\log n}=n$ subsets. Thus, the number of partitions in F is at most n^2.

F is good:

Let $p:[1\dots n]\to[0\dots w]$ be such that $\sum p_i\geq k$ and v a vertex of the tree. The *weight* of v under p, *weight*(v), is the sum of the weights of the elements in $dom(v)$. We concentrate on a distinguished path p_{heavy}: the path that starts at the root and each time goes to the heavier son.

We go along the path p_{heavy} from the root to the leaf and we label the vertices as being good or bad, as follows. Suppose we labeled the first $i-1$ vertices $v_1,\dots,v_{i-1}$ of the path p_{heavy} as being good or bad and declared t of them as being good (i may be 1 and t may be 0). Say, the t good vertices are $v_{g_1},\dots,v_{g_t}$. Let p_j be the middle point of $dom(v_{g_j})$, $j=1,\dots,t$. Denote by q the middle point of $dom(v_i)$, and by a and b

$(a<q<b)$ the partition points in $\{p_1,\ldots,p_t\}$ that are closest to q (if no partition point smaller (or bigger) than q exists we take the end point). Let us denote by t_r the number of partition points greater than q ($t_r \geq 0$), and similarly t_l is the number of partition points smaller than q. We call v_i good if:

- v_{i+1} (the heavy son of v_i) is the left son of v_i and $weight([\lceil q\rceil \ldots b)) \geq z_{l-t_r}$, or
- v_{i+1} (the heavy son of v_i) is the right son of v_i and $weight([a\ldots \lfloor q \rfloor)) \geq z_{t_l+1}$.

The t partition points $\{p_1,\ldots,p_t\}$ partition $[1\ldots n]$ into $t+1$ blocks. Furthermore, t of these blocks $B_1,\ldots,B_t$ will never be partitioned again. We call these blocks *inactive*. At each time there is only one *active* block. Also notice that the inactive blocks cover a prefix and a suffix of $[1\ldots n]$, and therefore we know exactly what their final index should be. Let us denote by k_j the weight of the j'th block from the left. We know that $k_j \geq z_j$. Next we show that at any stage the remaining weight $k - \sum_{j=1}^{t} weight(B_j)$ is big.

Lemma 4. *After finding t good vertices $k - \sum_{j=1}^{t} weight(B_j)$ is at least $\frac{k}{2^t} - \sum_{j=1}^{t_l} z_j - \sum_{j=l-t_r+1}^{l} z_j$.*

Proof. By induction on t.
The case $t=0$ $(t_l = t_r = 0)$ is trivial.
For $t=1$ we look at the first good vertex $v=v_i$. Since $v=v_i$ is the first good vertex, we get that the left and right flanks (the prefix and suffix outside $dom(v)$) have weight less than z_l and z_1 respectively. W.l.o.g. let us assume v_{i+1} is the left son of v_i. Then $B_1 = [q\ldots b]$, and since v is good it has weight at least z_1. However, its weight is also bounded from above by $\frac{weight(dom(v))}{2} + z_1$, since it is composed of the lighter half $[q\ldots c]$ of $dom(v)$ that can contribute at most $weight(dom(v))/2)$, and the remaining region $[c\ldots n]$ that weights less than z_1. Hence, the remaining weight is at least as required.
Now, assume for t and let us prove for $t+1$. Again, let us consider the t'th good vertex $v=v_i$. Say q is the middle point of $dom(v)$ and that a and b $(a<q<b)$ are the partition points in $\{p_1,\ldots,p_t\}$ that are closest to q (if no partition point smaller (or bigger) than q exists we take the end point). W.l.o.g. we assume v_{i+1} is the right son of v_i.
By induction we know that even ignoring $B_1,\ldots,B_t$ we still have at least $\frac{k}{2^t}$ minus the sum of the corresponding z_j weight. Let us now find the weight of the new added block B_{t+1}. By an argument similar to the case $t=1$, we see that its weight is at most half of what was left, plus a new

z_j corresponding the index of the new block. Thus, what is left when we remove the weight of B_{t+1} is at least as stated. ∎

Claim 8. *There are at least $l-1$ good vertices in p_{heavy}.*

Proof. Suppose there are only $t<l-1$ good vertices. This means that we have t inactive blocks $B_1,\dots,B_t$, t_l covering a prefix $[1\dots a]$, and t_r covering a postfix $[b\dots n]$. The remaining block $B=[a+1\dots b-1]$ is unable to support a new block neither in the left, nor in the right. So, $w(B)< z_{t_l+1}+z_{l-t_r}+w$. However, by Lemma 4, $w(B)$ is at least $\frac{k}{2^t}-\sum_{j=1}^{t_l} z_j - \sum_{j=l-t_r+1}^{l} z_j$. Thus, $\sum_i z_i+w>\frac{k}{2^t}\geq\frac{k}{2^l}$. A contradiction. ∎

Thus, these $l-1$ good vertices define a partition into l blocks such that the i'th block has weight at least z_i.

∎

6. The basic somewhere random extractor

Imagine being an extractor. You are given a string $x\in\{0,1\}^n$ that has large min-entropy. The first question you might ask yourself is "which of the bits of X is "more" random?". It turns out that instead of measuring the surprise of the i'th bit in the random source X, an even better idea is to consider the surprise of the i'th bit in the *given string*, i.e., to consider $q_i=\Pr(X_i=x_i\,|\,X_1=x_1,\dots,X_{i-1}=x_{i-1})$ as our surprise measure. This idea originates in the work of [6].

When taking this as our surprise measure we can see that if X has high min-entropy, almost all strings $x\in X$ have many surprising bits. This can be viewed as giving weights to the n bits, the weight of the i'th bit corresponds to its amount of "surprise", that add up together to something large. At the bit level we do not know which bit has high weight. However, at the block level we can use the small family of segmentations, and almost by definition, one of the segmentations in our small family must be good in the sense that it partitions $[1\dots n]$ into surprising blocks. This is the same as saying that the resulting blocks form a somewhere random source, which we already know how to handle. Thus trying all the possible segmentations (and there aren't too many of them) we know one of them will work and give us an almost uniform distribution. This idea, of trying all possible segmentations from a small fixed family, comes directly from the [11] paper, but the implementation here is more direct than in [6,11] and results in a simpler and stronger analysis.

Lemma 5. *Suppose*

- $\frac{k}{2^l} - \sum_{i=1}^{l} z_i \geq w = \log(\frac{n}{\epsilon})$. *From Lemma 3 we know there exists an explicit family F of segmentations of $[1 \ldots n]$ into l blocks that is $(k, [z_1, \ldots, z_l])$ good for any $p : [1 \ldots n] \to [0, w]$, and the size of the family F is at most n^2. Now further assume that,*
- *There is an explicit $(z'_1 = z_1 - \log(\frac{|F|}{\epsilon}), \ldots, z'_l = z_l - \log(\frac{|F|}{\epsilon}), \epsilon)$ block-wise extractor $E_\pi : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$.*

Then there is an explicit (k, ϵ) somewhere random extractor $E : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^{|F|}$.

Proof.

Let X be a distribution over $\{0,1\}^n$ with $H_\infty(X) \geq k$ and let $x \in X$. We define the somewhere random extractor:

The somewhere random extractor $S(x,y)$

Input $x \in \{0,1\}^n$.
Truly random string: $y \in \{0,1\}^d$.
Output: $S(x,y)$ has $|F|$ outputs indexed by the elements $\pi \in F$. For $\pi \in F$ the output B_π is $E_\pi(x,y)$, i.e., the block-wise extractor E_π applied on x partitioned by π.

We call an $x \in X$ "rare" if there is some i such that $\Pr(X_i = x_i \mid x_{i-1}, \ldots, x_1) \leq \frac{\epsilon}{n}$. It is easy to verify that $\Pr(x \text{ is rare}) \leq n\frac{\epsilon}{n} = \epsilon$. If x is rare we let $Y = 0$ (i.e., we failed).

For a non-rare $x \in X$ denote $q_i \overset{\text{def}}{=} \Pr(X_i = x_i \mid X_{i-1} = x_{i-1}, \ldots, X_1 = x_1)$. Clearly $\prod q_i = \Pr(X = x) \leq 2^{-k}$. Define $p = p(x) : [1 \ldots n] \to [0, w]$ by $p_i = \log(\frac{1}{q_i})$. It can be easily checked that $\sum p_i \geq k$, and $0 \leq p_i$. Furthermore, since x is not rare $p_i(x) \leq \log(\frac{n}{\epsilon}) = w$. Therefore, there is at least one partition in F that is good for p, and let us fix one such π. Let $Y = Y(x) = \pi$. Note that the weight function $p = p(x)$ and the segmentation $Y = Y(x)$ depend on x.

Denote by X_i^π the distribution of X when restricted to B_i^π, the i'th segment of π. Similarly for a string $b \in \{0,1\}^n$, b_i^π denotes the i'th block of b under the segmentation π. We now show that:

Claim 9. *For any $\delta > 0$, if $\Pr(Y = \pi) \geq \delta$ then $((X_1^\pi \circ \ldots \circ X_l^\pi) \mid Y = \pi)$ is a $(z_1 - \log(\frac{1}{\delta}), \ldots, z_l - \log(\frac{1}{\delta}))$ block-wise source.*

Proof. For $1 \leq i \leq l$ let $b_1^\pi, \ldots, b_i^\pi$ be such that they can be extended to some b with $Y(b) = \pi$.

Since $Y(b) = \pi$ we have that under the weight function $p = p(b)$, the weight of B_i^π is at least z_i. Consequently:

$$\Pr(X_i^\pi = b_i^\pi \mid {X_1}^\pi = b_1^\pi, \ldots, X_{i-1}^\pi = b_{i-1}^\pi) =$$

$$\prod_{j \in B_i^\pi} \Pr(X_j = b_j | X_1 = b_1, \ldots, X_{j-1} = b_{j-1}) =$$

$$\prod_{j \in B_i^\pi} q_j = 2^{-\sum_{j \in B_i^\pi} p_j} \leq 2^{-z_i}$$

Therefore,

$$\Pr(X_i^\pi = b_i^\pi | X_1^\pi = b_1^\pi, \ldots, X_{i-1}^\pi = b_{i-1}^\pi, Y = \pi)$$

$$\leq \frac{\Pr(X_i^\pi = b_i^\pi | X_1^\pi = b_1^\pi, \ldots, X_{i-1}^\pi = b_{i-1}^\pi)}{\Pr(Y = \pi)} \leq \frac{2^{-z_i}}{\delta}$$

And the claim follows.

Therefore, if we pick $\delta = \frac{\epsilon}{|F|}$, we get that whenever $\Pr(Y = \pi) \geq \frac{\epsilon}{|F|}$ we have that $(X_1^\pi \circ \ldots \circ X_l^\pi \,|\, Y = \pi)$ is a $(z_1', \ldots, z_l')$ block-wise source $(z_i' = z_i - \log(\frac{1}{\delta}))$, and therefore $(B_\pi \,|\, Y = \pi)$ is ϵ close to uniform.

Finally, for every π with $\Pr(Y = \pi) \leq \frac{\epsilon}{|F|}$ and every x with $Y(x) = \pi$ we redefine $Y(x)$ to be zero. It is now easy to see that the modified Y is an appropriate selector, and B is a somewhere random source. ∎

6.1. Proof of Theorem 5

Srinivasan and Zuckerman proved:

Theorem 6. [12] *There is a constant $C_{sz} > 1$ such that for every $k \geq \log n$ and $\epsilon \geq 2^{-k}$ there is a (k, ϵ) extractor $F : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ with:*

- *$m = k + d - \varDelta$ where $\varDelta = 2\log\frac{1}{\epsilon} + O(1)$, and*
- *$d = C_{sz}k$.*

Notice that the extractor has optimal entropy loss $2\log\frac{1}{\epsilon} + O(1)$. The extractor uses, however, $\varTheta(k)$ truly random bits instead of the optimal $O(\log(\frac{n}{\epsilon}))$. We are now ready to prove Theorem 5.

Proof (of Theorem 5). We use Lemma 5. We set

- $z_l' = O(\log(\frac{n}{\epsilon}))$, $t_l = C_{sz} z_l'$ and,
- $t_{i-1} = z_i' + t_i - \varDelta$, $z_{i-1}' = \frac{t_{i-1}}{C_{sz}}$

where l is the first integer such that $\sum z'_i \geq \Omega(\log^3 n \log \frac{1}{\epsilon})$. Now $z'_{i-1} = \frac{t_{i-1}}{C_{sz}} = \frac{z'_i + t_i - \Delta}{C_{sz}} = \frac{(z'_i - \Delta) + C_{sz} z'_i}{C_{sz}} = (1 + \Omega(1)) z'_i$. It follows that $l = O(\log\log(\frac{n}{\epsilon}))$. Lemma 5 tells us that we need $\frac{k}{2^l} - \sum(z'_i + \log \frac{n^2}{\epsilon}) \geq \log(\frac{n}{\epsilon})$, i.e., we need $k \geq poly\log(\frac{n}{\epsilon})$.

We now need to show a $(z'_1, \ldots, z'_l, \epsilon)$ block-wise extractor. Such a block-wise extractor was given in [12]. For completeness we sketch it here. Say $B_1, \ldots, B_l$ is the $(z'_1, \ldots, z'_l)$ block-wise source. Let F be the extractor of Theorem 6. We let $E(x_1, \ldots, x_l; y) = F(x_1, \ldots, F(x_{l-1}, F(x_l, y)) \ldots)$, i.e., we use F on the blocks $B_l, \ldots, B_1$ (and in that order), with the truly random seed for the current application taken to be the output of the previous application. For the correctness, we refer to [12]. The number of truly random bits used is $t_l = O(\log(\frac{n}{\epsilon}))$. The number of output bits is $\Omega(\sum z_i) = \Omega(\log^3 n \log \frac{1}{\epsilon})$ because at each application we lose at most $\Delta \leq \frac{z_i}{2}$ entropy. The error term is $l\epsilon$. Starting with $\epsilon' = \frac{\epsilon}{l}$ we get the result. ∎

We notice that the block-wise extractor does not lose much entropy. The only place where we lose $poly\log(\frac{n}{\epsilon})$ entropy, is in Lemma 5 that guarantees us only a 2^{-l} fraction of the entropy in the source.

7. Further work and open problems

In this paper we constructed efficient somewhere random extractors for sources having any min-entropy. In particular this reduces the problem of finding explicit *general* extractors to that of finding explicit extractors *for somewhere random sources*. Extractors for somewhere random sources are called "mergers" in [5] where some explicit constructions with non-optimal degree are presented. It will be very interesting (and useful) to have a direct (and hopefully efficient) construction for mergers. Another open problem is reducing the entropy loss to $O(\log n)$. This will bring the explicit constructions for depth two super-concentrator and *a* expanding graphs, to almost optimal.

Recently, Impagliazzo, Shaltiel and Wigderson [3,4] constructed an extractor with an optimal seed length that works for any given min-entropy k. [3,4] use complicated recursion that builds on an extractor construction of Trevisan [13]. The [3,4] result improve on our work in that they construct extractors rather than somewhere random extractors. On the other hand, the entropy loss in their construction is $k^{\Omega(1)}$ while it is only $poly\log(\frac{n}{\epsilon})$ in ours.

Very recently, Reingold, Shaltiel and Wigderson [8] came up with another extractor construction that works for any min-entropy k. Their construction improves the [6,12] results by using, among others, techniques from [5] that are similar to those used in this paper. Yet, even this construction has $\Omega(k)$ entropy loss ([8] state their result only for a constant ϵ).

Acknowledgments. I would like to thank the anonymous referee for many helpful comments that greatly improved the presentation of the paper.

References

[1] B. Chor and O. Goldreich: Unbiased bits from sources of weak randomness and probabilistic communication complexity, *SIAM Journal on Computing*, **17(2)** (1988), 230–261.

[2] O. Goldreich and D. Zuckerman: Another proof that BPP ⊆ PH (and more), in: *Electronic Colloquium on Computational Complexity, technical reports,* 1997.

[3] R. Impagliazzo, R. Shaltiel and A. Wigderson: Near-optimal conversion of hardness into pseudo-randomness, in: *Symposium on Foundations of Computer Science (FOCS),* 1999.

[4] R. Impagliazzo and R. Shaltiel and A. Wigderson: Extractors and pseudo-randomn generators with optimal seed-length, in: *Proceedings of the Thirty-second Annual ACM Symposium on the Theory of Computing,* 2000.

[5] N. Nisan and A. Ta-Shma: Extracting randomness: A survey and new constructions, *Journal of Computer and System Sciences*, **58,** 1999.

[6] N. Nisan and D. Zuckerman: Randomness is linear in space, *Journal of Computer and System Sciences,* **52** (1996), 43–52.

[7] R. Raz, O. Reingold and S. Vadhan: Extracting all the randomness and reducing the error in trevisan's extractors, in: *ACM Symposium on Theory of Computing (STOC),* 1999.

[8] O. Reingold, R. Shaltiel and A. Wigderson: Extracting randomness via repeated condensing, in: *IEEE Symposium on Foundations of Computer Science (FOCS),* 2000.

[9] J. Radhakrishnan and A. Ta-Shma: Bounds for dispersers, extractors, and depth-two superconcentrators, *SIAM Journal on Discrete Mathematics,* **13(1)** (2000), 2–24.

[10] M. Sipser: Expanders, randomness, or time versus space, *Journal of Computer and System Sciences*, **36** (1988).

[11] M. Saks, A. Srinivasan and S. Zhou: Explicit OR-dispersers with polylogarithmic degree, *Journal of the ACM,* **45** (1998), 123–154.

[12] A. Srinivasan and D. Zuckerman: Computing with very weak random sources, *SIAM Journal on Computing,* **28(4)** (1999), 1433–1459.

[13] L. Trevisan: Construction of extractors using pseudo-random generators (extended abstract), in: *ACM Symposium on Theory of Computing (STOC),* 1999.

[14] A. Wigderson and D. Zuckerman: Expanders that beat the eigenvalue bound: Explicit construction and applications, in: *Proceedings of the Twenty-Fifth Annual ACM Symposium on the Theory of Computing,* 1993, 245–251, San Diego, California,

[15] D. Zuckerman: General weak random sources, in: *Proc. 31st Ann. IEEE Symp. on Foundations of Computer Science,* 1990, 534–543.

[16] D. Zuckerman: On Unapproximable Versions of *NP*-Complete Problems, *SIAM Journal on Computing*, **25(6)** (1996), 1293–1304.

[17] D. Zuckerman: Randomness-optimal sampling, extractors, and constructive leader election, in: *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing,* 1996, 286–295, Philadelphia, Pennsylvania.

[18] D. Zuckerman: Simulating BPP using a general weak random source, *Algorithmica,* **16(4/5)** (1996), 367–391.

Amnon Ta-Shma

Computer Science Department,
Tel-Aviv University, Israel 69978.
amnon@post.tau.ac.il