Dispersers with logarithmic entropy loss

Noam Parzanchevski

August 4, 2019

Abstract

Fixing an error in the proof of [TS02] and improving upon the results therein, we construct dispersers over 2^n vertices for any min-entropy $k + O\left(\log^3 \frac{k}{\varepsilon}\right)$ where $k \geq c \log \frac{n}{\varepsilon}$ (for some constant c) of degree $\text{poly}\ \frac{\text{n}}{\varepsilon}$ and $O\left(\log \frac{k}{\varepsilon}\right)$ entropy loss

1 Introduction

The focus on the construction of pseudorandom objects has stood as a cornerstone in the theory of randomness since its earliest days. Some of the most important results in the field of derandomization were achieved via the use of pseudorandom objects such as *pseudorandom generators, expander graphs* and *extractors* (see, for example, [Nis94, Rei08]). Pseudorandom objects are objects that can be constructed explicitly (i.e., deterministically and efficiently) but exhibit certain aspects that appear to behave randomly. This work deals with the construction of *dispersers*, low-degree bipartite graphs that have large vertex expansion (id est, if V_1, V_2 are the partition of the graph vertices then every large enough subset of V_1 covers almost all of V_2).

Definition 1. *(disperser) A* $(D = 2^d)$*-left regular bipartite graph*

$$G = (V_1 = [N = 2^n], V_2 = [M = 2^m], E)$$

is a $(K = 2^k, \varepsilon)$*-disperser, if for every subset* $W \subseteq V_1$ *of size at least* K *it holds that* $|\Gamma(W)| \geq (1-\varepsilon)M$ *where* $\Gamma(W)$ *is the set of neighbors of* W*. The* entropy loss *of the disperser is* $\Delta(G) = k + d - m$*.*

1.1 Deterministic construction and low degree

If we do not limit ourselves to an explicit construction and maintaining a low degree then constructing a bipartite graph $G = (V_1, V_2, E)$ with large vertex expansion is an easy task:

- Without a limitation on the degree of the graph, we can simply construct the complete bipartite graph. This graph obviously has optimal expansion but its degree is extremely high

- On the other hand, non-explicitly, one can show using the probabilistic method that for any $\varepsilon > 0$ and k there exists a (k, ε)-disperser of degree $\Theta(\frac{n}{\varepsilon})$ and that this is also the lower bound for such a graph (see [RTS00])

Our goal will be to present a *deterministic* construction of dispersers with *low degree*.

1.2 Extraction and entropy loss

It is clear that what we want is a disperser with k, ε and degree as small as possible. Another important and less obvious measure of the effectiveness of our construction is the *entropy loss* we incur. In order to explain this it makes more sense to switch to a different perspective and examine *Extractors*, which are stronger pseudorandom objects that are easily reducible to *Dispersers*. Extractors are functions which take as input a *flawed* random source X (in the sense that X is distributed over n bit but has $k < n$ min-entropy) and an additional small input of d truly uniform random bits and output m bits that are ε-close to uniform.

In this setting we "invest" d bits of randomness and "harvest" m bits of entropy out of the total $k + d$ bits of entropy in the system. We call the margin between the two our entropy loss $\Delta \overset{\text{def}}{=} k + d - m$.

Coming back to dispersers, let $G = (V_1, V_2, E)$ be a bipartite graph of degree $D = 2^d$ where $|V_1| = N = 2^n$. Identify the vertices in V_1 with n-bit strings by an arbitrary enumeration $\phi : \{0,1\}^n \to V_1$ and enumerate the edges drawn from each vertex $v \in V_1$ by $0, \ldots, D - 1$. With this approach, consider any source $X \subseteq \{0,1\}^n$ with k bits of entropy. By definition, for any $x \in \text{Supp}\,(X) : \Pr[X = x] \leq 2^{-k}$ and so $|\text{Supp}\,(X)\,| \geq 2^k$. Thus, we can think of X as a random variable distributed over a subset $W \subseteq V_1$ of size $\geq 2^k$ by letting $W = \{\phi(x) \mid x \in \text{Supp}\,(X)\}$. Additionally, a truly random seed $y \in \{0,1\}^d$ can be thought of as a random selection of an edge.

The entropy loss is in this case is the logarithmically scaled margin $\log \frac{|W| \cdot D}{|V_2|} = k + d - \log |V_2|$, comparing the size of W plus the degree of the graph (our "investment") and the size of V_2 (our "reward"). The dispersers we will construct will incur a polylogarithmic entropy loss.

In order to achieve the above, we will construct a third pseudorandom object which will turn out to be stronger than a disperser but weaker than an extractor - a *somewhere random extractor*.

1.3 Previous works and our contribution

Dispersers were first defined in [Sip88] where a probabilistic proof was given to show that a random bipartite graph $G = (V_1, V_2, E)$ where $|V_1| = n, |V_2| = 2^{\sqrt{\log n}}$ of degree $2 \log n$ is a $(2^{\sqrt{\log n}}, 1/2)$-disperser with high probability. However, the task of providing an explicit construction remained open.

In [SZ99], an explicit construction of a $(n^{\gamma}, 1/2)$-disperser was given for any constant $\gamma > 0$ with a $\mathrm{poly\ n}$ degree and $\Omega(n^{\gamma})$ entropy loss. In [SSZ98], a construction was given with similar parameters for any $\varepsilon = \mathrm{poly}\ \frac{1}{\mathrm{n}}$.

In later works, [TSUZ07] showed how to construct (k, ε)-dispersers for any k and with entropy loss $\Omega(\log n)$ if ε is constant and [GUV09] constructed dispersers for any k and $\varepsilon > 0$ with entropy loss $\Omega(k)$.

From a different perspective, non-explicit results from [RTS00] show that for any k, ε there exist dispersers of degree $\Theta(\frac{n}{\varepsilon})$ with entropy loss only $\log \log \frac{1}{\varepsilon} + O(1)$ and that this is also a lower bound on the entropy loss.

The work we present in this paper is built upon the construction given in [TS02], where (k, ε)-dispersers of $\mathrm{poly}\ \frac{\mathrm{n}}{\varepsilon}$ degree were constructed for $k = \mathrm{poly}\ \log \frac{\mathrm{n}}{\varepsilon}$ and any $\varepsilon > 0$ with a $\mathrm{poly}\ \log \frac{\mathrm{n}}{\varepsilon}$ entropy loss. The parameters of [TS02] improve upon previous results both in terms of the entropy loss and in supporting any $k \geq \mathrm{poly}\ \log \frac{\mathrm{n}}{\varepsilon}$ and any error rate. However, the proof in [TS02] contained an error.

In this work we fix the error, streamline the construction and improve the parameters we achieve. Namely, for any $\varepsilon > 0, k \geq c \log \frac{n}{\varepsilon}$ for some absolute constant c we construct $\left(k + O\left(\log^3 \frac{k}{\varepsilon}\right), \varepsilon\right)$-dispersers of $\mathrm{poly}\ \frac{\mathrm{n}}{\varepsilon}$ degree and $O\left(\log \frac{k}{\varepsilon}\right)$ entropy loss.

The following table summarizes previous results discussed above:

Degree D	Min entropy $k = \log K$	Error ε	Entropy loss	Reference and notes
$D = \mathrm{poly\ n}$	$k = n^{\Omega(1)}$	$\varepsilon = 1/2$	$n^{\Omega(1)}$	[SZ99]
$D = \mathrm{poly\ n}$	$k = n^{\Omega(1)}$	$\varepsilon \geq \mathrm{poly}\ \frac{1}{\mathrm{n}}$	$n^{\Omega(1)}$	[SSZ98]
$D = \mathrm{poly\ n}$	Any k	Constant	$3 \log n + O(1)$	[TSUZ07]
$D = \mathrm{poly}\ \frac{\mathrm{n}}{\varepsilon}$	Any k	Any ε	$\Omega(k)$	[GUV09]
$D = \mathrm{poly}\ \frac{\mathrm{n}}{\varepsilon}$	$k = \mathrm{poly}\ \log \frac{\mathrm{n}}{\varepsilon}$	Any ε	$\mathrm{poly}\ \log \frac{\mathrm{n}}{\varepsilon}$	[TS02], see below
$D = \mathrm{poly}\ \frac{\mathrm{n}}{\varepsilon}$	$k \geq c \log \frac{n}{\varepsilon}$	Any ε	$O\left(\log \frac{k}{\varepsilon}\right)$	This paper
$D = \Theta(\frac{n}{\varepsilon})$	Any k	Any ε	$\log \log \frac{1}{\varepsilon} + O(1)$	[RTS00], non-explicit
$D = \Theta(\frac{n}{\varepsilon})$	Any k	Any ε	$\log \log \frac{1}{\varepsilon} + O(1)$	[RTS00], lower bound

As mentioned above, the construction in [TS02] contained an error which was pointed to us by Arkadev Chattopadhyay, Michael Langberg, Shi Li and Atri Rudra. Namely, in the proof of Claim 9 in the original paper an erroneous inequality was used. Our result fixes this issue and improves upon [TS02] in several respects:

- Correctness: as mentioned, first and formost, we fix the error in the original paper. A detailed review of the error and the fix employed is available in section 5.1.1.

- Parameters: while the original construction requires $k = \mathrm{poly}\ \log\frac{\mathrm{n}}{\varepsilon}$ bits of entropy and proves a $\mathrm{poly}\ \log\frac{\mathrm{n}}{\varepsilon}$ entropy loss, we actually compute the exponents and shows that if $k \geq c\log\frac{n}{\varepsilon}$ for some constant c then $k + O\left(\log^3\frac{k}{\varepsilon}\right)$ bits of entropy is sufficient and that the entropy loss we incur is only $O\left(\log\frac{k}{\varepsilon}\right)$.

- Simplification: the original construction of [TS02] required the composition of two types of extractors. The correctness proof for the pseudorandom properties of this composition was fairly arduous. By using a different composition and state of the art extractors we manage to bypass this composition and provide a simpler, streamlined construction. A detailed explanation of this process is available in section 5.1.2

Finally, it was brought to our attention by Ronen Shaltiel that if we allow for a polylogarithmic entropy loss then there is a simple construction of dispersers with similar parameters to ours and an entropy loss of order $O\left(\log k \cdot \log\frac{k}{\varepsilon}\right)$. We present this proof in appendix D.

1.4 Main results: asserting the results of [TS02]

While our end goal is to construct a disperser graph, this work deals mainly with the construction of a somewhere random extractor with logarithmic seed length and polylogarithmic entropy loss:

Theorem 1. *For any $n \geq k, \varepsilon > 0$ there exists an $\left(k + O(\log^3\frac{n}{\varepsilon}\right), \varepsilon)$-somewhere random extractor*

$$S : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^{n^3}$$

with $d = O(\log\frac{n}{\varepsilon})$ and $m = k - 2\log\frac{1}{\varepsilon} - O(1)$.

By applying an initial condensing step using the [GUV09] lossless condenser, we achieve even better results:

Theorem 2. *There exists a constant c such that for any $\varepsilon > 0$ and $c\log\frac{n}{\varepsilon} < k \leq n$ there exists a (k_1, ε)-somewhere random extractor*

$$S : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^{O(k_1^3)}$$

where $k_1 = k + O\left(\log^3\frac{k}{\varepsilon}\right), d = O\left(\log\frac{n}{\varepsilon}\right), m = k_1 + O\left(\log\frac{n}{\varepsilon}\right)$. Furthermore, the entropy loss of S is $\Delta(S) = O\left(\log\frac{k}{\varepsilon}\right)$

Using Lemma 5, the above immediately implies the following:

Theorem 3. *There exists a constant c such that for any $\varepsilon > 0$ and $c\log\frac{n}{\varepsilon} < k \leq n$ there exists a $(2^{k_1}, \varepsilon)$-disperser*

$$G = (V_1 = \{0,1\}^n, V_2 = \{0,1\}^m, E)$$

of degree $D = \mathrm{poly}\ \frac{\mathrm{n}}{\varepsilon}$ where $k_1 = k + O\left(\log^3\frac{k}{\varepsilon}\right)$ and $m = k_1 + O\left(\log\frac{n}{\varepsilon}\right)$. Furthermore, the entropy loss of G is $\Delta(G) = O\left(\log\frac{k}{\varepsilon}\right)$

1.5 Organization

In what follows, section 2 contains definitions, preliminary information and reductions between the various pseudorandom objects presented in the paper. A top down view of the proof is presented in section 3. Sections 4-5 contain a detailed proof of the main theorem.

2 Preliminaries and background

In this section we begin by defining the pseudorandom objects we will use for our construction.

2.1 Statistical Distance and Sources

Throught the paper, we will denote the uniform distribution over $\{0,1\}^d$ by U_d. When d is clear from context we will sometimes simply denote it by U.

Definition 2. *(Statistical distance) Let X and Y be two random variables distributed over the same universe Λ. The* Statistical distance *between X, Y is:*

$$|X - Y| = \frac{1}{2}|X - Y|_1 = \frac{1}{2}\sum_{\lambda\in\Lambda}|\Pr[X = \lambda] - \Pr[Y = \lambda]|$$

If $|X - Y| \leq \varepsilon$ we say that X is ε-close to Y.

We record the well known fact that statistical distance between distributions can only shrink:

Fact 1. *Let A, B be two distributions over the same universe Λ. For any function $f : \Lambda \to \Lambda'$:*

$$|f(A) - f(B)| \leq |A - B|$$

And use it to establish the following useful corollary:

Corollary 2. *Let A, B be two distributions over the same universe Λ such that $|B - A| \leq \varepsilon_1$ and $f : \Lambda \to \Lambda'$ a function such that $|f(A) - U| \leq \varepsilon_2$, then by the triangle inequality*

$$|f(B) - U| \leq |f(B) - f(A)| + |f(A) - U| \leq \varepsilon_1 + \varepsilon_2$$

Definition 3. *(Weak source) Let X be a distribution over $\{0,1\}^n$. The* min-entropy *of X is $H_\infty(X) = \log\frac{1}{\max_x \Pr[X=x]}$. We say X is a k-source if $H_\infty(x) \geq k$, or, equivalently,* $\Pr[X = x] \leq 2^{-k}$ *for every $x \in X$.*

Definition 4. *[CG88] (Block-wise source) Suppose X is a random variable taking values from $\{0,1\}^n$, and π is a partition of $[1..n]$ 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, $X = X_1^\pi \circ \ldots \circ X_\ell^\pi$ where the X_i^π are possibly correlated.*

We say X is a $(\pi, z_1, \ldots, z_\ell)$ block-wise source, if for every $x \in \{0,1\}^n$ for which $\Pr[X = x] > 0$ and for every $1 \le i \le \ell$ 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 $(z_1, \ldots, z_\ell)$ block-wise source.

We will strive to partition our input random variables in a way which guarantees sufficient min entropy in each block. We formalize this requirement in the following definition:

Definition 5. *(Segmentations) A* segmentation *of $[n]$ to ℓ blocks is a partition of $[n]$ into ℓ blocks $B_1 = [1..b_1], B_2 = [b_1 + 1..b_2], \ldots, B_\ell = [b_{\ell-1} + 1..n]$. A family F of segmentations of $[n]$ into ℓ blocks is $(k, [z_1, \ldots, z_\ell], w)$ good, if for any weight function $p : [1..n] \to [0..w]$ with $\sum_i p(i) \ge k$ there is at least one segmentation $\pi \in F$ that partitions $[1..n]$ into blocks $B_1, \ldots, B_\ell$ such that for every $1 \le j \le \ell : \sum_{i \in B_j} p(i) \ge z_j$.*

2.2 Extractors

Next, we define the basic pseudorandom object we will utilize in our construction:

Definition 6. *(Extractor) Let $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$.*

- *Let F be a family of distributions over $\{0,1\}^n$. We say E is a (F, ε)-extractor, if for every $X \in F$, $|E(X, U_d) - U_m| \le \varepsilon$.*
- *We say E is a (k, ε)-extractor, if it is an (F, ε) extractor for the family F of all k-sources.*
- *We say E is an $(\pi, [z_1, \ldots, z_l], \varepsilon)$-block wise extractor if it is an (F, ε) extractor for the family of all $(\pi; z_1, \ldots, z_l)$ block-wise sources. E may depend on π (and $z_1, \ldots, z_l$ and ε), and when we want to emphasize this we write E_π.*

Adapting a construction of [SZ99] we know that:

Lemma 3. *There exist constants $c > 1$, $\alpha \in (1, 2)$ such that for all ℓ, if $z_{\ell-i} \ge \alpha^i c \log \frac{n}{\varepsilon}$ there exists a $([z_0, z_1, \ldots, z_\ell], \varepsilon)$ block wise extractor:*

$$F : \{0,1\}^n \times \{0,1\}^{r_\ell} \to \{0,1\}^m$$

with $m \ge \Omega(\alpha^\ell \log \frac{n}{\varepsilon})$ and $r_\ell = O(z_\ell + \log n)$. If $z_1 = \Omega(\log^2 \frac{n}{\varepsilon})$ then there exists such an extractor with $m \ge z_0 - 2\log \frac{1}{\varepsilon} - O(1)$

We give the proof, for completeness, in appendix A.

2.3 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 7. *[NTS99] (Somewhere random source) $B = (B_1, \ldots, B_b)$ is a b-block (m, ε) somewhere random source if each B_i is a random variable over $\{0,1\}^m$ and there is a random variable Y over $[0, \ldots, b]$ such that:*

- *For every $i \in [1, \ldots, b] : \Pr[Y = i] > 0 \implies |(B_i | Y = i) - U_m| \leq \varepsilon$.*
- $\Pr[Y = 0] \leq \varepsilon$.

We call Y a 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 ε.

We define a somewhere random extractor to be a function whose output is a somewhere random source.

Definition 8. *(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 $S(X, U_d) = B_1 \circ \ldots \circ B_b$ is obtained by picking $x \in X, y \in U_d$ and computing $S(x, y)$.*

We say S is a (k, ε) somewhere random extractor *if for every distribution X which is a k-source, $\{B_1, \ldots, B_b\}$ is a b-block (m, ε) somewhere random source.*

Similar to our definition of dispersers, given a (k, ε)-extractor or somewhere random extractor E with uniform seed length d whose output length is m, the *entropy loss* of the extractor is $\Delta(E) = k + d - m$.

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 (possibly correlated) 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.

2.4 Reductions

As stated earlier, we will construct a disperser using a somewhere random extractor. The following lemmas show the relation between extractors, somewhere random extractors and dispersers.

2.4.1 Extractors to dispersers

We first present a straightforward reduction from extractors to dispersers:

Lemma 4. *Given $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$, a (k,ε)-extractor, one can construct a $(K = 2^k, \varepsilon)$-disperser $G = (V_1, V_2, E)$ where $|V_1| = N = 2^n, |V_2| = M = 2^m$ of degree $D = 2^d$*

Proof. Identify V_1 with the set $\{0,1\}^n$ and V_2 with $\{0,1\}^m$. For each $x \in \{0,1\}^n, y \in \{0,1\}^d$ such that $E(x,y) = z$ add the edge (x,z) to G. Clearly, this is a D regular bipartite graph.

Now, let $W \subseteq V_1$ be a set of size $|W| \geq K$. The uniform distribution over W is a k-source, thus $E(W, U_d)$ is ε-close to uniform and misses at most an ε-fraction of $\{0,1\}^m$. It follows that $|\Gamma(W)| \geq (1-\varepsilon)M$ □

2.4.2 Somewhere random extractors to dispersers

Every (k,ε)-extractor is also a somewhere random extractor outputting a single block with a selector Y such that $\Pr[Y = 0] = 0$. Thus in a sense a somewhere random extractor is weaker then an extractor. The following lemma shows that a somewhere random extractor is still a stronger object than a disperser:

Lemma 5. *Given $S : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^b$, a (k,ε)-somewhere random extractor, one can construct a $(K = 2^k, \varepsilon)$-disperser $G = (V_1, V_2, E)$ where $|V_1| = N = 2^n, |V_2| = M = 2^m$ of degree $b \cdot D = b \cdot 2^d$*

Proof. Identify V_1 with the set $\{0,1\}^n$ and V_2 with $\{0,1\}^m$. For each $x \in \{0,1\}^n, y \in \{0,1\}^d$ write $S(x,y) = S_1(x,y) \circ \cdots S_b(x,y) = z_1 \circ \cdots \circ z_b$. Add the edges $(x,z_1), \ldots, (x,z_b)$ to G. It is easy to see that this is a $b \cdot D$ regular bipartite graph.

Again, let $W \subseteq V_1$ be a set of size $|W| \geq K$. As $S(W, U_d)$ is a somewhere random source we have a selector function Y for it. Now, let $i \in [b]$ such that $\Pr[Y = i] > 0$. By a similar argument to that in Lemma 4 even if we restrict the edges of G to the ith block of the output (that is, we connect x and $S_i(x,y)$ for any $x \in \{0,1\}^n, y \in \{0,1\}^d$) the set $\Gamma(W)$ misses at most an ε-fraction of the vertices in V_2 □

3 Top-down structure of the proof

First, given a source X and a set of requirements $z_1, \ldots, z_\ell$, we give a combinatorial lemma which shows that if we are willing to "pay a little extra" in the entropy placed on X then there is a small family of segmentations F of the interval $[n]$ such that at least one such segmentation will split the input into ℓ blocks, where the ith block has at least z_i bits of entropy. The extra cost in this stage accounts for the polylogarithmic entropy loss of our final somewhere random extractor.

Theorem 4. *Suppose* $k \geq \left(\sum_{j=1}^{\ell} z_j + w\right) \cdot 2^{\ell} + z_0 + w$ *for some positive values* k, ℓ, n, z_j, w. *Then there is a family* F *of segmentations of* $[1..n]$ *into* $\ell + 1$ *blocks that is* $(k, [z_0, z_1, \ldots, z_\ell], w)$ *good such that the size of* F *is at most* n^3.

We give the proof of Theorem 4 in Section 4.

Next, we show that applying appropriate block-wise extractors over a family of segmentations gives a somewhere-random extractor:

Theorem 5. *Suppose for* $n, \varepsilon = \varepsilon(n) > 0$ *and* $k = k(n)$ *the following holds:*

- *There exists an explicit family* F *of segmentations of* $[1..n]$ *into* $\ell = \ell(n)$ *blocks that is* $(k, [z_1, \ldots, z_\ell], w)$ *good, for* $w = \log(\frac{4n}{\varepsilon})$.
- *There exists an explicit* $((z'_1, \ldots, z'_\ell), \varepsilon)$ *block-wise extractor* $E_\pi : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *for* $z'_j = z_j - 2\log\frac{1}{\varepsilon} - \log 4|F| - \log 2\ell$.

Then there exists an explicit (k, ε)*-somewhere random extractor:*

$$S : \{0,1\}^n \times \{0,1\}^d \to (\{0,1\}^m)^{|F|}$$

Plugging the block-wise extractor of Lemma 3 and the family of segmentations of Theorem 4 into Theorem 5 gives Theorem 1. Applying a condensing step on the somewhere random extractor of Theorem 1 gives Theorem 2. For completeness, we give the calculations for Theorem 1 in Appendix C and for Theorem 2 in appendix B.

4 A small family of segmentations

Lemma 6. *Suppose* $\frac{k}{2^\ell} - \sum_{j=1}^{\ell} z_j \geq w$ *for some positive values* k, ℓ, n, z_j, w. *Then there is a family* F *of segmentations of* $[n]$ *into* ℓ *blocks that is* $(k, [z_1, \ldots, z_\ell], w)$ *good such that the size of* F *is at most* n^2.

Proof. We assume wlog that n is a power of 2 (otherwise we can add dummy elements of weight zero), and consider the elements in $[1, ..., n]$ as the leaves of a full binary tree. For an inner node v we denote by $\mathrm{dom}(v)$ the leaves of the subtree rooted in v (note that this is a consecutive interval). We define for both leaves and intervals the appropriate weight function:

- $\mathrm{wt}(v) = \sum_{i \in \mathrm{dom}(v)} p(i)$
- $\mathrm{wt}(B_j) = \sum_{i \in B_j} p(i)$

We now give the construction of F:

- For each path from the root to a leaf $P = (v_1, ..., v_{\lg n})$ take all subsets of size $\ell - 1$ from P

- For each such subset, each v_j puts a partition point in the middle of $\mathrm{dom}(v_j)$ (e.g. - the root partitions the interval into $[1..\frac{n}{2}]$, $[\frac{n}{2}+1..n]$). These $\ell-1$ partition points thus induce an ℓ-partition of $[1..n]$

Notice that there are exactly n paths in the tree, so

$$|F| = n \cdot \binom{\lg n}{\ell-1} \leq n \cdot 2^{\lg n} = n^2$$

Now fix a weight function $p : [1..n] \to [0..w]$ such that $\sum_i p(i) = k$. We will show that there exists a $\pi \in F$ such that $\forall j : \sum_{i \in B_j} p(i) \geq z_j$. We run Algorithm 1 and construct a set *Good* of *good* vertices.

```
Algorithm 1: Vertex labeling
Input: $T, p, (z_1, ..., z_\ell)$
Output: $Good$
Initializations: $v \leftarrow \mathrm{root}(T)$, $Good = \{1, n\}$, $t_l = t_r = 0$
while $v$ is not a leaf do
    $q \leftarrow mid(v)$
    Let $a, b \in Good$ be closest to $q$ s.t $a < q < b$
    if $\mathrm{heavy\_son}(v) = \mathrm{left}(v)$ and $\mathrm{wt}([q+1..b]) \geq z_{\ell-t_r}$ then
        $Good \leftarrow Good \cup \{v\}$; $t_r \leftarrow t_r + 1$
    else if $\mathrm{heavy\_son}(v) = \mathrm{right}(v)$ and $\mathrm{wt}([a..q]) \geq z_{t_{l+1}}$ then
        $Good \leftarrow Good \cup \{v\}$; $t_l \leftarrow t_l + 1$
    $v \leftarrow \mathrm{heavy\_son}(v)$
end
return $Good$
```

For every $t \geq 2$, After Algorithm 1 finds t good vertices $Good_t$, the vertices in $Good_t$ induce a partition into $t-1$ blocks. Of these blocks, $t-2$ blocks $B_1, \ldots, B_{t-2}$ will never be touched again and we call them *inactive blocks*. We order these blocks by the order in which they were created. The only block which could change (the *active* block) is the one spanned by the heavy son of the latest good vertex. We further denote by w_j the required weight of block B_j, i.e., if B_j is the t_ℓ block from the left than $w_j = z_{t_\ell}$, and if it is the t_r block from the right than $w_j = z_{\ell-t_r+1}$. We claim:

Claim 7. *Using the above notation, for every $t \geq 2$, it holds that*

$$k - \sum_{j=1}^{t-2} \mathrm{wt}(B_j) \geq \frac{k}{2^{t-2}} - \sum_{j=1}^{t-2} w_j$$

Proof. By induction on t. $t = 2$ is trivial. Assume for t and prove for $t+1$. Suppose the $t+1$ block was created when v_i was added to *Good*. Wlog, assume $v_{i+1} = \mathrm{heavy\_son}(v_i)$ is the right son of v_i. Let $a < q < b$ be as in Algorithm 1. The active block is exactly the leaves spanned by the heavy son v_{i+1}, thus, by the induction hypothesis,

$$\begin{aligned}
\mathrm{wt}(v_{i+1}) &= k - \sum_{j=1}^{t-1} \mathrm{wt}(B_j) \\
&= k - \sum_{j=1}^{t-2} \mathrm{wt}(B_j) - \mathrm{wt}(B_{t-1}) \\
&\geq \frac{k}{2^{t-2}} - \sum_{j=1}^{t-2} w_j - \mathrm{wt}(B_{t-1})
\end{aligned}$$

Next we note that $\mathrm{wt}(B_{t-1})$ is composed of the lighter half of $\mathrm{dom}(v_i)$ and an interval which was lighter than w_{t-1}, so $\mathrm{wt}(B_{t-1}) \leq \frac{\mathrm{wt}(v_i)}{2} + w_{t-1}$ and therefore

$$\begin{aligned}
\mathrm{wt}(v_{i+1}) &\geq \frac{k}{2^{t-2}} - \sum_{j=1}^{t-2} w_j - \left(\frac{\mathrm{wt}(v_i)}{2} + w_{t-1}\right) \\
&= \frac{k}{2^{t-2}} - \sum_{j=1}^{t-1} w_j - \frac{\mathrm{wt}(v_i)}{2}.
\end{aligned}$$

Thus, $\frac{\mathrm{wt}(v_i)}{2} + \mathrm{wt}(v_{i+1}) \geq \frac{k}{2^{t-2}} - \sum_{j=1}^{t-1} w_j$. As v_{i+1} is the heavy son v_i we have $\mathrm{wt}(v_{i+1}) \geq \frac{\mathrm{wt}(v_i)}{2}$ and $2 \cdot \mathrm{wt}(v_{i+1}) \geq \frac{\mathrm{wt}(v_i)}{2} + \mathrm{wt}(v_{i+1})$. Hence,

$$\mathrm{wt}(v_{i+1}) \geq \frac{1}{2}\left(\frac{k}{2^{t-2}} - \sum_{j=1}^{t-1} w_j\right) \geq \frac{k}{2^{t-1}} - \sum_{j=1}^{t-1} w_j$$

concluding the proof of the claim. □

Next, we claim that there are at least $\ell + 2$ good vertices in P_{heavy}. Suppose not. Then there are at most $t \leq \ell + 1$ good vertices and let v_{heavy} be the heavy son of the last good vertex. Wlog, let us assume $t = \ell + 1$. We know that the inactive blocks defined by these vertices $B_1..B_{t-2}$ have t_l blocks covering a prefix $[1..a]$ and t_r covering a suffix $[b..n]$ and $t_\ell + t_r = t - 2 \leq \ell - 1$, and the remaining block $B = [a + 1..b - 1]$ which is spanned by the leaves of v has

$$\mathrm{wt}(B) \;\; = \;\; \mathrm{wt}(v_{heavy}) = k - \sum_{j=1}^{\ell-2} \mathrm{wt}(\ B_j) \geq \frac{k}{2^{\ell-1}} - \sum_{j=1}^{\ell-1} w_j.$$

Let z be the element of $\{z_1, \ldots, z_\ell\}$ not covered by $\{w_1, \ldots, w_{\ell-1}\}$. As $\frac{k}{2^\ell} - \sum_j z_j \geq w$, we have $\frac{k}{2^{\ell-1}} = 2\frac{k}{2^\ell} \geq 2\sum_j w_j + 2w$, and

$$\mathrm{wt}(B) \;\; \geq \;\; 2z + w. \tag{1}$$

Now suppose the heavy path ends at the leaf q. Since we do not have any further good vertices both $\mathrm{wt}([a+1..q-1]) < z$ and $\mathrm{wt}([q+1..b-1]) < z$. Hence, $\mathrm{wt}(B) = \mathrm{wt}(v_{heavy}) = \mathrm{wt}([a+1..b-1]) \leq z + w(q) + z$. But $w(q) < w$, a contradiction to Eq (1). Thus we have at least $\ell + 2$ good vertices in P_{heavy} which defined ℓ blocks $B_1, \ldots, B_\ell$ with $\mathrm{wt}(B_j) \geq w_j$ as desired □

Having that, we prove Theorem 4:

Proof. As $p(j) \leq w$ for any j there exists an index i such that $\sum_{j=0}^{i} p(j) \geq z_0$ and $\sum_{j=i+1}^{n} p(j) \geq k$. By the lemma, we know that there exists a family F' of segmentations of $i+1, \ldots, n$ into ℓ blocks which is $(k, [z_1, \ldots, z_\ell], w)$-good. Thus, for any weight function p we have a $\pi \in F'$ which partitions p properly on the interval $[i{+}1..n]$ for $z_1, \ldots, z_\ell$, that is to say that if $\pi = B_1, \ldots B_\ell$ then for every $1 \leq j \leq \ell$ we have $\sum_{i \in B_j} p(i) \geq z_j$. As additionally $\sum_{j=0}^{i} p(j) \geq z_0$, we get that $B_0 = [1..i], B_1, \ldots B_\ell$ is a proper partition for p on the entire interval $[1..n]$. Therefore we can simply output for each $\pi \in F'$ and each $i \in n$ the segmentation $B_0 = [1..i] \circ \pi$. If we denote by $F'_{[a..b]}$ the segmentation family guaranteed for the interval $[a..b]$ we get:

$$|F| = \sum_{j=1}^{n} |F'_{[j..n]}| \leq n \cdot |F'_{[1..n]}| \leq n^3$$

□

5 Using a family of segmentations to construct a somewhere-random extractor

We now show how to use a block wise extractor, "extra" entropy and a family F of segmentations to construct a somewhere random extractor which outputs $|F|$ blocks:

Proof. (of Theorem 5) We define

$$S : \{0,1\}^n \times \{0,1\}^d \rightarrow (\{0,1\}^m)^{|F|}$$

by simply running our somewhere random extractor over all partitions in F and outputing each one as a block. For a random variable X taking values $x \in \{0,1\}^n$ and $\tau \in F$ we denote by X_i^τ and x_i^τ the ith block of X and x (respectively) as partitioned by τ. Our somewhere random extractor works as follows:

Algorithm 2: The Somewhere Random Extractor S

```
Input: $x \in \{0,1\}^n$ [input], $y \in \{0,1\}^d$ [seed]
for $\tau \in F$ do
    return $E_\pi(x_1^\tau \circ \cdots \circ x_\ell^\tau, y)$
end
```

To see that S is a somewhere random extractor, let X be the distribution over $\{0,1\}^n$ with $H_\infty(X) \geq k$. We need some notation:

- We say $x \in X$ is *rare* if $\Pr[X_i = x_i | x_1, \ldots, x_{i-1}] \le \frac{\varepsilon}{4n}$ for some $i \in [n]$. Notice that by a union bound $\Pr[x \text{ is rare}] \le n \cdot \frac{\varepsilon}{4n} = \frac{\varepsilon}{4}$.
- For a non-rare $x \in X$ we define

$$\begin{aligned} q_x(i) &= \Pr[X_i = x_i | X_1 = x_1, \ldots X_{i-1} = x_{i-1}], \text{ and,} \\ p_x(i) &= \log \frac{1}{q_x(i)}. \end{aligned}$$

- Also set $\delta = \frac{\varepsilon}{4|F|}$.

For a non-rare x, for every $i \in [n]$, $p_x(i) \le \log(\frac{4n}{\varepsilon}) = w$, and therefore the weight function p_x is $p_x : [1..n] \to [0..w]$. Also, for every x, $\prod q_x(i) = \Pr[X = x] \le 2^{-k}$ and therefore $\sum p_x(i) \ge k$. Thus, by assumption, for every non-rare x there exists at least one partition π_x in F that is good for p_x. Let

$$\Pi = \Pi(x) = \begin{cases} 0 & x \text{ is rare} \\ \pi_x & \text{Otherwise} \end{cases}$$

And let

- $\mathrm{Rare}_{0,\pi}(b)$ be a boolean random variable getting value 1 iff $\Pi(b) = \pi$ and $\Pr[\Pi(x) = \pi] < \delta$.
- $\mathrm{Rare}_{j,\pi}(b)$ for $j \in [\ell - 1]$, be a boolean random variable getting value 1 iff $\Pi(b) = \pi$ and $\Pr[\Pi(x) = \pi \mid X_1^\pi = b_1^\pi, \ldots, X_j^\pi = b_j^\pi] < \frac{\varepsilon}{2\ell} \cdot \Pr[\Pi = \pi]$.

Let

$$\Pi'(x) = \begin{cases} \pi & \Pi(x) = \pi \neq 0 \text{ and } \mathrm{Rare}_{j,\pi}(x) = 0 \text{ for every } 0 \le j \le \ell - 1. \\ 0 & \text{Othewrwise} \end{cases}$$

We advise the reader to think of Π as an initial selector and Π' as a more refined selector for S. The following claims show that if the initial selector chose a "not unlikely" partition then it will "survive" the refined selector with high probability.

Claim 8. *If* $\Pr[\Pi = \pi] > \delta$ *then* $\Pr[\Pi' = \pi] > (1 - \frac{\varepsilon}{2}) \cdot \Pr[\Pi = \pi]$.

Proof. Fix $\pi \neq 0$ such that $\Pr[\Pi = \pi] = \gamma > \delta$. By definition of Π',

$$\begin{aligned} \Pr_{x \in X}[\Pi'(x) = \pi] &= \Pr_{x \in X}[\Pi(x) = \pi \text{ and } \textstyle\bigwedge_{j=0}^{\ell-1} \mathrm{Rare}_{j,\pi}(x) = 0] \\ &\ge \Pr[\Pi = \pi] - \Pr_{x \in X}[\exists_{j=1}^{\ell-1} \mathrm{Rare}_{j,\pi}(x) = 1] \\ &\ge \Pr[\Pi = \pi] - \sum_{j=0}^{\ell-1} \Pr_{x \in X}[\mathrm{Rare}_{j,\pi}(x) = 1]. \end{aligned}$$

As $\Pr[\Pi = \pi] > \delta$ we have $\mathrm{Rare}_{0,\pi}(x) = 0$ for all x with $\Pi(x) = \pi$. Now, define for every $j \in [\ell - 1]$ the set of j-block prefixes that can be extended to a word whose jth block is rare:

$$R_{\pi,j} = \{c_1^\pi \circ \cdots \circ c_j^\pi \mid \exists c_{j+1}^\pi \circ \cdots \circ c_\ell^\pi : \Pi(c) = \pi \ \wedge \ \mathrm{Rare}_{j,\pi}(c) = 1\}$$

Note that for any $c_1^\pi \circ \cdots \circ c_j^\pi \in R_{\pi,j}$ we have by definition

$$\Pr_{x\in X}[\Pi(x) = \pi \mid x_1^\pi = c_1^\pi, \ldots, x_j^\pi = c_j^\pi] < \frac{\varepsilon}{2\ell} \cdot \Pr_{x\in X}[\Pi(x) = \pi]$$

we now claim that $\Pr_{x\in X}[\mathrm{Rare}_{j,\pi} = 1] \leq \frac{\varepsilon}{2\ell} \Pr_{x\in X}[\Pi(x) = \pi]$. This follows as $\Pr_{x\in X}[\mathrm{Rare}_{j,\pi} = 1]$ is a convex sum of the probability that a prefix in $R_{\pi,j}$ is extended to a word whose jth block is rare. Indeed

$$\begin{aligned}
\Pr_{x\in X}[\mathrm{Rare}_{j,\pi}(x) = 1] = & \sum_{c_1^\pi \circ \cdots \circ c_j^\pi \in R_{\pi,j}} \Pr_{x\in X}[x_1^\pi = c_1^\pi, \ldots, x_j^\pi = c_j^\pi] \\
& \Pr_{x\in X}[x_{j+1}^\pi = c_{j+1}^\pi, \ldots, x_\ell^\pi = c_\ell^\pi, \Pi(x) = \pi, \mathrm{Rare}_{j,\pi}(x) = 1 \mid x_1^\pi = c_1^\pi, \ldots, x_j^\pi = c_j^\pi] \\
& \leq \sum_{c_1^\pi \circ \cdots \circ c_j^\pi \in R_{\pi,j}} \Pr_{x\in X}[x_1^\pi = c_1^\pi, \ldots, x_j^\pi = c_j^\pi] \Pr_{x\in X}[\Pi(x) = \pi \mid x_1^\pi = c_1^\pi, \ldots, x_j^\pi = c_j^\pi] \\
& \leq \left(\sum_{c_1^\pi \circ \cdots \circ c_j^\pi \in R_{\pi,j}} \Pr_{x\in X}[x_1^\pi = c_1^\pi, \ldots, x_j^\pi = c_j^\pi] \right) \\
& \cdot \max_{d_1^\pi \circ \cdots \circ d_j^\pi \in R_{\pi,j}} \Pr_{x\in X}[\Pi(x) = \pi \mid x_1^\pi = d_1^\pi, \ldots, x_j^\pi = d_j^\pi] \\
& \leq \max_{d_1^\pi \circ \cdots \circ d_j^\pi \in R_{\pi,j}} \Pr_{x\in X}[\Pi(x) = \pi \mid x_1^\pi = d_1^\pi, \ldots, x_j^\pi = d_j^\pi] \\
& < \frac{\varepsilon}{2\ell} \cdot \Pr_{x\in X}[\Pi(x) = \pi]
\end{aligned}$$

And thus,

$$\Pr[\Pi' = \pi] \ \geq \ \gamma - \sum_{j=1}^{\ell-1} \gamma \cdot \frac{\varepsilon}{2\ell} \ \geq \ \gamma(1 - \varepsilon \sum_{j=1}^{\ell} \frac{1}{2\ell}) \ = \ \gamma(1 - \frac{\varepsilon}{2}).$$

□

Next, we show that the above implies:

Corollary 9.

$$\Pr[\Pi' = \pi | \Pi = \pi] \geq 1 - \frac{\varepsilon}{2}$$

Proof. We want to show that the "survival" probability of the refined selector is high, that is to say that if $\Pi = \pi \neq 0$ then with significantly high probability $\Pi' = \pi$ as well. Note that by the law of total probability

$$\Pr[\Pi' = \pi] = \sum_{\tau} \Pr[\Pi' = \pi \wedge \Pi = \tau]$$

Since for any $\tau \neq \pi$ we have $\Pr[\Pi' = \pi \wedge \Pi = \tau] = 0$ we get that $\Pr[\Pi' = \pi] = \Pr[\Pi' = \pi \wedge \Pi = \pi]$, thus:

$$\Pr[\Pi' = \pi | \Pi = \pi] = \frac{\Pr[\Pi' = \pi \wedge \Pi = \pi]}{\Pr[\Pi = \pi]} \geq 1 - \frac{\varepsilon}{2}$$

□

We now prove that the refined selector Π' is indeed a selector function for S:

Claim 10. *If* $\Pr[\Pi' = \pi] > 0$ *then* $(X_1^\pi \circ \cdots \circ X_\ell^\pi | \Pi' = \pi)$ *is a* $(z_1', \ldots, z_\ell')$ *block-wise source and therefore* $(E_\pi(X, U) \mid \Pi' = \pi)$ *is* ε*-close to uniform.*

Proof. Fix $1 \leq j \leq \ell$ and $b_1^\pi, \ldots, b_j^\pi$ that can be extended to some b with $\Pi(b) = \pi$. Since $\Pi(b) = \pi$ we have that under the weight function $p = p_b$ the weight of B_j^π is at least z_j. Consequently:

$$\begin{aligned} \Pr[X_j^\pi = b_j^\pi | X_1^\pi = b_1^\pi, \ldots, X_{j-1}^\pi = b_{j-1}^\pi] &= \prod_{i \in B_j^\pi} \Pr[X_i = b_i | X_1 = b_1, \ldots, X_{i-1} = b_{i-1}] \\ &= \prod_{i \in B_j^\pi} q_b(i) = 2^{-\sum_{i \in B_j^\pi} p_b(i)} \leq 2^{-z_j}. \end{aligned}$$

Let $\gamma = \Pr[\Pi = \pi]$. As $\Pi'(b) = \pi \neq 0$ we know that $\gamma \geq \delta$ and $\mathrm{Rare}_{j-1,\pi}(b) = 0$, i.e.,

$$\Pr[\Pi = \pi | X_1^\pi = b_1^\pi, \ldots, X_{j-1}^\pi = b_{j-1}^\pi] \geq \frac{\varepsilon}{2\ell} \cdot \gamma$$

Hence,

$$\begin{aligned} \Pr[X_j^\pi = b_j^\pi | X_1^\pi = b_1^\pi, \ldots, X_{j-1}^\pi = b_{j-1}^\pi, \Pi = \pi] &\leq \frac{\Pr[X_j^\pi = b_j^\pi | X_1^\pi = b_1^\pi, \ldots, X_{j-1}^\pi = b_{j-1}^\pi]}{\Pr[\Pi = \pi | X_1^\pi = b_1^\pi, \ldots, X_{j-1}^\pi = b_{j-1}^\pi]}, \\ &\leq \frac{2^{-z_j}}{(\varepsilon/2\ell) \cdot \gamma} \leq \frac{2^{-z_j}}{\varepsilon\delta} \cdot 2\ell = 2^{-z_j'}, \end{aligned}$$

where we have used $\Pr[A \mid B, C] \leq \frac{\Pr[A \mid B]}{Pr[C \mid B]}$. The claim follows □

Next, we show that $\Pi' \neq 0$ with high probability:

Claim 11. $\Pr[\Pi' = 0] \leq \varepsilon$

Proof. If $\Pr[\Pi = \pi] = \gamma > \delta$, then it follows from Corollary 9 that $\Pr[\Pi' = 0 | \Pi = \pi] \leq \frac{\varepsilon}{2}$. Letting $F_{<\delta} = \{\pi \in F \mid \Pr[\Pi = \pi] < \delta\}$ and $F_{\geq\delta} = F \backslash F_{<\delta}$, we have:

$$\begin{aligned}
\Pr[\Pi' = 0] &\leq \Pr[\Pi = 0] + \Pr[\Pi' = 0 \mid \Pi \neq 0] \\
&\leq \frac{\varepsilon}{4} + \sum_{\pi \in F} \Pr[\Pi = \pi] \Pr[\Pi' = 0 \mid \Pi = \pi] \\
&\leq \frac{\varepsilon}{4} + \sum_{\pi \in F_{<\delta}} \Pr[\Pi = \pi] + \sum_{\pi \in F_{\geq\delta}} \Pr[\Pi = \pi] \Pr[\Pi' = 0 \mid \Pi = \pi] \\
&\leq \frac{\varepsilon}{4} + \sum_{\pi \in F_{<\delta}} \delta + \sum_{\pi \in F_{\geq\delta}} \Pr[\Pi = \pi] \cdot \frac{\varepsilon}{2} \\
&\leq \frac{\varepsilon}{4} + \delta|F| + \frac{\varepsilon}{2} = \varepsilon.
\end{aligned}$$

□

By Claim 10 we know that Π' is a proper selector function for S. By Claim 11 we know that $\Pr[\Pi' = 0] \leq \varepsilon$. We conclude that S is indeed a (k, ε)-somewhere random extractor. □

5.1 A note on the changes from [TS02]

5.1.1 Fixing the error

In the original paper, a step used in the proof of Claim 10 stated that

$$\Pr[X_j^\pi = b_j^\pi | X_1^\pi = b_1^\pi, \ldots, X_{j-1}^\pi = b_{j-1}^\pi, \Pi = \pi] \leq \frac{\Pr[X_j^\pi = b_j^\pi | X_1^\pi = b_1^\pi, \ldots, X_{j-1}^\pi = b_{j-1}^\pi]}{\Pr[\Pi = \pi]}$$

relying on the erroneous inequality $\Pr[A \mid B, C] \leq \frac{\Pr[A \mid B]}{\Pr[C]}$ for arbitrary random variables A, B, C. The correct inequality is $\Pr[A \mid B, C] \leq \frac{\Pr[A \mid B]}{\Pr[C \mid B]}$.

To mend the argument, we have introduced a sequence of auxiliary random variables $\mathrm{Rare}_{j,\pi}(b)$ for $j \in [\ell - 1]$ which informally indicate that the selector Π chose an unlikely segmentation π given the prefix $X_1^\pi, \ldots, X_j^\pi$. We say $b \in \{0,1\}^n$ is unlikely (for a given $j \in [\ell - 1]$) if $\Pi(b) = \pi$ and the probability to get π given the $j - 1$ prefix of b is at most $\varepsilon_j \cdot \Pr(\Pi = \pi)$. We choose the ε_j such that $\sum_j \varepsilon_j$ is $\frac{1}{2}\varepsilon$.

The downside of the fix is an increase in the weight of z_j'. We can think of the discrepancy ratio $\log \frac{z_j'}{z_j}$ as the "extra entropy" we need to pay in order to get our somewhere random extractor. In our case, this amounts to $O(\log \frac{1}{\varepsilon} + \log n + \log \ell)$ whereas the previous construction had a discrepancy of $O(\log \frac{1}{\varepsilon} + \log n)$. As $\ell \leq n$, this is essentially the same (up to constant factors).

5.1.2 Streamlining the construction

Informally speaking, Theorem 5 states that given a block wise extractor that works for block wise sources with entropy $(z_1, \ldots, z_\ell)$, if we are given a source with "a little extra entropy" $k \geq \sum_{i=1}^{\ell} z_i$ then one can construct a (k, ε)-somewhere random extractor. By plugging the parameters of Lemma 3 we essentially use a $(z_0, \ldots, z_\ell)$ block wise source where z_0 contains the bulk of our entropy and all other blocks which contain $(z_1, \ldots, z_\ell)$ bits of entropy can be thought of as a supply of "auxiliary entropy". Theorem 4 then allows us to extract z_0 bits of entropy with minimal entropy loss (as we only incur an entropy loss on blocks $1, \ldots, \ell$).

The corresponding proof in [TS02] (Theorem 3, the composition lemma) has a different structure. Using a somewhere random extractor S and an extractor E the following composition takes place:

1. For any segmentation π of the source X into $X_1 \circ X_2$ let:

   (a) $S_{1,\pi}, \ldots, S_{t,\pi} = S(X_2, U)$

   (b) $E_{1,\pi}, \ldots, E_{t,\pi} = E(X_1, S_{1,\pi}), \ldots, E(X_1, S_{t,\pi})$

   Where U is a truly random seed.

2. The output of the composition is

$$E \circ S(X) = E_{1,\pi_1}, \ldots, E_{t,\pi_1} \circ E_{1,\pi_2}, \ldots, E_{t,\pi_2} \circ \ldots, \circ E_{1,\pi_n}, \ldots, E_{t,\pi_n}$$

   I.e., for each segmentation π_i of X into $X_{[1,i]}, X_{[i+1,n]}$ output the composition of $E \circ S$ when X is segmented according to π_i

It is then proved in a fairly complicated and technical manner that $E \circ S$ is in itself a somewhere random extractor. Finally, the proof shows that one of the segmentations yields an X_1 with z_0 bits of entropy in X_1 and X_2 which is a $(z_1, \ldots, z_\ell)$ blocks wise source yielding the necessary extraction properties.

By extracting the entropy from all blocks including z_0 using a single somewhere random extractor we were able to simplify the construction and the proof of correctness of Theorem 5. By using state of the art extractors we were able to extract the entropy from the zeroth block with minimal entropy loss (see appendix A).

6 Acknowledgments

We thank Arkadev Chattopadhyay, Michael Langberg, Shi Li and Atri Rudra for pointing out the error in the proof of [TS02]. We thank Ronen Shaltiel for pointing out the simple proof of theorem 9.

References

[CG88] Benny Chor and Oded Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM Journal on Computing*, 17(2):230–261, 1988.

[GUV09] Venkatesan Guruswami, Christopher Umans, and Salil Vadhan. Unbalanced expanders and randomness extractors from parvaresh–vardy codes. *Journal of the ACM (JACM)*, 56(4):20, 2009.

[Nis94] Noam Nisan. RL in SC. *Computational Complexity*, 4(1):1–11, 1994.

[NTS99] Noam Nisan and Amnon Ta-Shma. Extracting randomness: A survey and new constructions. *Journal of Computer and System Sciences*, 58(1):148–173, 1999.

[Rei08] Omer Reingold. Undirected connectivity in log-space. *Journal of the ACM (JACM)*, 55(4):17, 2008.

[RTS00] Jaikumar Radhakrishnan and Amnon Ta-Shma. Bounds for dispersers, extractors, and depth-two superconcentrators. *SIAM Journal on Discrete Mathematics*, 13(1):2–24, 2000.

[Sip88] Michael Sipser. Expanders, randomness, or time versus space. *Journal of Computer and System Sciences*, 36(3):379–383, 1988.

[SSZ98] Michael Saks, Aravind Srinivasan, and Shiyu Zhou. Explicit OR-dispersers with polylogarithmic degree. *Journal of the ACM (JACM)*, 45(1):123–154, 1998.

[SZ99] Aravind Srinivasan and David Zuckerman. Computing with very weak random sources. *SIAM Journal on Computing*, 28(4):1433–1459, 1999.

[TS02] Amnon Ta-Shma. Almost optimal dispersers. *Combinatorica*, 22(1):123–145, 2002.

[TSUZ07] Amnon Ta-Shma, Christopher Umans, and David Zuckerman. Lossless condensers, unbalanced expanders, and extractors. *Combinatorica*, 27(2):213–240, 2007.

A Constructing the Block Wise Extractor

We now prove Lemma 3. In section A.1 we first show we can extract $m = \Omega(\alpha^\ell \log \frac{n}{\varepsilon})$, and then in section A.2 we show that if $z_1 = \Omega(\log^2 \frac{n}{\varepsilon})$ then m can be much larger. Finally, we compare the block wise extractor we give to the one constructed in [TS02] in section A.3.

A.1 The [SZ99] block wise extractor

Proof. (of Lemma 3, first part) Our proof will use the following extractor which is based on the improved Leftover Hash Lemma presented in [SZ99] :

Theorem 6. *(Lemma 3.2 in [SZ99]) There exists a constant $C_{sz} > 1$ such that for every $\varepsilon > 0, \Delta \geq 2\log\frac{1}{\varepsilon} + 2, k \geq \Delta$ there exists a (k, ε)-extractor*

$$F : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^{k-\Delta+d}$$

where $d = C_{sz}(\log n + k)$

We will also require the following technical claim, the proof of which is given in the next section:

Claim 12. *Let $c \geq 4$, $\alpha = 1 + \frac{1}{c \cdot C_{sz}}$ and denote by $f(i) = \alpha^i c \log\frac{n}{\varepsilon}$. Then for any integer $i \geq 0$:*

$$f(i) + C_{sz}(\log n + f(i)) - 2\log\frac{1}{\varepsilon} - 2 \geq C_{sz}\left(\log n + f(i+1)\right)$$

Set $c \geq 4$, $r_\ell = C_{sz}(\log n + z_\ell)$ and $\alpha = 1 + \frac{1}{c \cdot C_{sz}}$. We now show by induction on ℓ that we have a block wise extractor with output length at least $C_{sz}(\log n + \alpha^{\ell+1} c \log\frac{n}{\varepsilon})$. For the base case, $\ell = 0$, by assumption $z_0 \geq c\log\frac{n}{\varepsilon}$ and $r_\ell = C_{sz}(\log n + z_0)$ thus by Theorem 6 we have a (z_0, ε)-extractor

$$E : \{0,1\}^n \times \{0,1\}^{r_\ell} \to \{0,1\}^{m = z_0 + r_\ell - 2\log\frac{1}{\varepsilon} - 2}$$

where $m \geq C_{sz}(\log n + \alpha c \log\frac{n}{\varepsilon})$ by Claim 12.

Next, set $\varepsilon' = \frac{\varepsilon}{2}$ and let

$$E_\ell : \{0,1\}^n \times \{0,1\}^{r_\ell} \to \{0,1\}^{m_\ell}$$

be a $([z_0, \dots, z_\ell], \varepsilon')$ block wise extractor guaranteed by the induction hypothesis where $m_\ell \geq C_{sz}(\log n + \alpha^{\ell+1} c \log\frac{n}{\varepsilon'})$. Consider a $[z_0, \dots, z_\ell]$ block wise source X_ℓ where $z_{\ell-i} \geq \alpha^i c \log\frac{n}{\varepsilon'}$ and note that $|E_\ell(X_\ell, U_{r_\ell}) - U_{m_\ell}| \leq \varepsilon'$.

Now, let X be a z-source where $z \geq \alpha^{\ell+1} c \log\frac{n}{\varepsilon'}$. Again, by Theorem 6 we have a (z, ε') extractor

$$E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^{m' = z + d - 2\log\frac{1}{\varepsilon'} - 2}$$

where $d = C_{sz}(\log n + z)$. It follows by corollary 2 that

$$|E(X, E_\ell(X_\ell, U_{r_\ell})) - U_{m'}| \leq 2 \cdot \varepsilon' = \varepsilon$$

Finally,

$$m' \geq \alpha^{\ell+1} c \log\frac{n}{\varepsilon'} + C_{sz}(\log n + \alpha^{\ell+1} c \log\frac{n}{\varepsilon'}) - 2\log\frac{1}{\varepsilon'} - 2$$

which is at least $C_{sz}(\log n + \alpha^{\ell+2} c \log\frac{n}{\varepsilon'})$ by Claim 12. Letting $F(X \circ X_\ell, U_{r_\ell}) = E(X, E_\ell(X_\ell, U_{r_\ell}))$ be our block wise extractor, the claim follows □

Proof. (of Claim 12) A straightforward computation shows:

$$\begin{aligned}
&f(i) + C_{sz}(\log n + f(i)) - 2\log\frac{1}{\varepsilon} - 2 = \\
&C_{sz}\left(\log n + \left(1 + \frac{1}{C_{sz}}\right) f(i) - \frac{2\log\frac{1}{\varepsilon} + 2}{C_{sz}}\right) \geq \\
&C_{sz}\left(\log n + \left(1 + \frac{1}{C_{sz}}\right) f(i) - 3\log\frac{n}{\varepsilon}\right) \geq \\
&C_{sz}\left(\log n + \left(1 + \frac{1}{C_{sz}} - \frac{3}{c}\right) f(i)\right) = \\
&C_{sz}\left(\log n + \left(1 + \frac{c-3}{c\cdot C_{sz}}\right) f(i)\right) \geq \\
&C_{sz}\left(\log n + \left(1 + \frac{1}{c\cdot C_{sz}}\right) f(i)\right) = \\
&C_{sz}\left(\log n + f(i+1)c\log\frac{n}{\varepsilon}\right)
\end{aligned}$$

Where we use $c \geq 4$ for the last inequality □

A.2 Extracting all of the remaining entropy

Next, we complete the proof of Lemma 3 showing how to extract the remaining entropy in the first block with optimal loss.

Proof. (of Lemma 3, second part)

We now assume $z_1 = \Omega(\log^2 \frac{n}{\varepsilon})$, we want to extract the entropy from z_0 with optimal loss, and for that we use the following extractor, due to [GUV09]:

Theorem 7. *(Theorem 4.21 in [GUV09]) There exists a constant C_{guv} such that for all positive integers $n \geq k$ and all $\varepsilon > 0$, there is an explicit (k, ε)-extractor $F_{GUV} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ where $m = k + d - 2\log\frac{1}{\varepsilon} - O(1)$ and $d = \log n + C_{guv}(\log k \cdot \log(\frac{k}{\varepsilon}))$.*

Now, let

$$E : \{0,1\}^n \times \{0,1\}^{r_\ell} \to \{0,1\}^{m'=2\cdot C_{guv}\log^2\frac{n}{\varepsilon'}}$$

be a $([z_1, \dots, z_\ell], \varepsilon')$ block wise extractor guaranteed by the previous section (note: $m' \geq \log n + C_{guv}(\log k \cdot \log(\frac{k}{\varepsilon}))$ for any $k \leq n$), and let

$$F_{GUV} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$$

be the (z_0, ε')-extractor guaranteed by Theorem 7. As before, if X_ℓ is a $[z_1, \dots, z_\ell]$ block wise source and X is a z_0 source we use corollary 2 to see that:

$$|F_{GUV}(X, E(X_\ell, U_{r_\ell})) - U_m| \leq 2\cdot\varepsilon' = \varepsilon$$

and $m = z_0 + d - 2\log\frac{1}{\varepsilon} - O(1) > z_0 - 2\log\frac{1}{\varepsilon} - O(1)$ as required. Again, letting $F(X \circ X_\ell, U_{r_\ell}) = F_{GUV}(X, E_\ell(X_\ell, U_{r_\ell}))$ be our block wise extractor, the claim follows □

A.3 Comparing the block wise extractor to [TS02]

In the terminology of Lemma 3, the block wise extractor shown in [TS02] (which is based on the construction of [SZ99]) does not deal with the the block z_0, namely, it is a $([z_1, \ldots, z_\ell], \varepsilon)$ block wise extractor:

$$F : \{0,1\}^n \times \{0,1\}^{r_\ell} \to \{0,1\}^m$$

with $m \geq \Omega(\alpha^\ell \log\frac{n}{\varepsilon})$ and $r_\ell = O(z_\ell + \log n)$. The task of extracting the bulk of the entropy which is placed in the zeroth block was then delegated to a different part of the proof (see section 5.1.2 for more details).

By using the [GUV09] extractor on the zeroth block we were able to simplify our construction as detailed in section 5.1.2.

B Applying the initial condensing step

We will use the following theorem of [GUV09]:

Theorem 8. *(Theorem 4.3 in [GUV09] with $\alpha = 1, k_{\max} = k$) For any $k \leq n$ and $\varepsilon > 0$ there exists a $k \to_\varepsilon k + d$ lossless converter*

$$C : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$$

where $d = 2\log\frac{nk}{\varepsilon} + O(1)$ and $m = 2(d + k)$.

Id est, given a k-source X over n bits, the output $C(X, U_d)$ is ε-close to some $(k + d)$-source over m bits.

With this, we're ready to prove theorem 2.

Proof. (of Theorem 2)

Let $\varepsilon > 0, n \geq k \geq c\log\frac{n}{\varepsilon}$ for some contant to be specified later. We will need to compose a condenser with a somewhere random extractor.

B.1 The condenser

Let $\varepsilon_1 = \varepsilon/2, k_1 = k + c_1\log^3\frac{k}{\varepsilon_1}, d_1 = c_2\log\frac{nk_1}{\varepsilon_1} + O(1)$ for some constants c_1, c_2 such that

- $c > 2c_2$ (observe that as $k_1 \leq n$ we know that $k > d_1$ and therefore also $k_1 > d_1$)
- $c_2 > 2$

And let

$$C_1 : \{0,1\}^n \times \{0,1\}^{d_1} \to \{0,1\}^{2(k_1+d_1)}$$

be a $k_1 \to_{\varepsilon_1} k_1 + d_1$ lossless condenser guaranteed by theorem 2.

B.2 The somewhere random extractor

Let $\varepsilon_2 = \varepsilon/2, k_2 = k_1 + d_1, n_2 = 2(k_1+d_1), d_2 = O(\log \frac{n_2}{\varepsilon_2})$ such that there exists a (k_2, ε_2)-somewhere random extractor

$$S_2 : \{0,1\}^{n_2} \times \{0,1\}^{d_2} \to (\{0,1\}^{m_2})^{n_2^3}$$

where $m_2 = k_2 - 2\log\frac{1}{\varepsilon_2} - O(1) = k_2 - 2\log\frac{1}{\varepsilon} - O(1)$ guaranteed by theorem 1

B.3 The composition

We now claim that $S : \{0,1\}^n \times \{0,1\}^{d_1+d_2} \to (\{0,1\}^{m_2})^{n_2^3}$ defined by

$$S(X, U_{d_1} \circ U_{d_2}) = S_2(C_1(X, U_{d_1}), U_{d_2})$$

is a (k_1, ε)-somewhere random extractor.

To see this, let X be a k_1-source over n bits. By the properties of C_1, we know that $C_1(X, U_{d_1})$ is a distribution over $2(k_1+d_1)$ bits which is $(\varepsilon/2)$-close to a $k_1 + d_1$ source.

We want to show that a $k_2 = (k_1+d_1)$-source has sufficient entropy for S_2. Id est, we need to show that

$$k_1 + d_1 = k + c_1 \log^3 \frac{k}{\varepsilon_1} + d_1 \geq k + c_{\text{sre}} \log^3 \frac{n_2}{\varepsilon_2}$$

for the constant c_{sre} required by the somewhere random extractor. As $\varepsilon_1 = \varepsilon_2$ and $n_2 = 2(k_1+d_1) = 2k_2$ we can clearly choose c_1 such that the above holds.

Since $C_1(X, U_{d_1})$ is ε_1-close to a source with sufficient entropy for S_2, we see by corollary 2 that $S(X, U_{d_1} \circ U_{d_2})$ is a $(k_1, \varepsilon_1 + \varepsilon_2 = \varepsilon)$-somewhere random extractor.

B.4 The parameters we achieve

The entropy we require is $k_1 = k + O(\log^3 \frac{k}{\varepsilon})$, the uniform seed we require has length $d_1 + d_2 = O(\log \frac{nk_1}{\varepsilon_1} + \log \frac{n_2}{\varepsilon_2}) + O(1) = O\left(\log \frac{n}{\varepsilon}\right)$ and the entropy loss we incur is

$$k_1 + d_1 + d_2 - m_2 = k_1 + d_1 + d_2 - (k_1 + d_1 - 2\log\frac{1}{\varepsilon} - O(1)) = d_2 + 2\log\frac{1}{\varepsilon} + O(1)$$

As $d_2 = O\left(\log \frac{n_2}{\varepsilon_2}\right)$ and $n_2 = 2(k_1+d_1) \leq 4k_1 = O\left(k + \log^3 \frac{k}{\varepsilon}\right)$ we see that the entropy loss Δ is of order $\Delta = O\left(\log\left(\frac{k+\log^3\frac{k}{\varepsilon}}{\varepsilon}\right)\right) + O(1)$. As $k + \log^3 \frac{k}{\varepsilon} =$

$O(k + \log^3 k + \log^3 \frac{1}{\varepsilon}) = O(k + \log^3 \frac{1}{\varepsilon})$ we bound the loss by

$$\Delta = O\left(\log\left(k + \log^3 \frac{1}{\varepsilon}\right) + \log \frac{1}{\varepsilon}\right) = O\left(\log k + \log \frac{1}{\varepsilon}\right) = O\left(\log \frac{k}{\varepsilon}\right)$$

Finally, the length of each block we output is $m_2 = k_2 - 2\log\frac{1}{\varepsilon} - O(1) = k_1 + d_1 - 2\log\frac{1}{\varepsilon} - O(1) = k_1 + O\left(\log\frac{n}{\varepsilon}\right)$ and the number of blocks we output is $n_2^3 = (2(k_1 + d_1))^3 \leq (4k_1)^3 = O(k_1^3)$

□

C Plugging the parameters for theorem 1

Proof. (of Theorem 1) We want to use Theorem 5 and for that we need both a somewhere random extractor and a family of segmentations. For the somewhere random extractor, we set the parameters as specified in the proof of Lemma 3: Let $\ell = \log\left(2 \cdot C_{guv} \cdot \log\frac{n}{\varepsilon}\right)$, for $0 \leq i < \ell$: $z'_{\ell-i} = 2^i c \log\frac{n}{\varepsilon}$ and $z'_0 = k$ where we think of k as the number of bits we wish to extract and $\sum_{i=1}^{\ell} z'_i$ as “auxiliary” bits.

Now, as $z'_1 = 2 \cdot C_{guv} \cdot \log^2 \frac{n}{\varepsilon}$, by Lemma 3 we have a $([z'_0, \ldots, z'_\ell], \varepsilon)$-block wise extractor:

$$E : \{0,1\}^n \times \{0,1\}^{r_\ell} \to \{0,1\}^m$$

with $m \geq k - 2\log\frac{1}{\varepsilon} - O(1)$.

Next, for $0 \leq i \leq \ell$ define $z_i = z'_i + 2\log\frac{1}{\varepsilon} + \log 4|F| + \log 2\ell$ and finally, let $k' = z_0 + w + \left(\sum_{i=1}^{\ell} z_i + w\right) \cdot 2^\ell$ and observe that by Theorem 4 we have a family F of size n^3 which is $(k', [z_0, \ldots, z_\ell], w)$ good where $w = \log(\frac{4n}{\varepsilon})$. Next, we claim that k', the amount of entropy which we will require for our final somewhere random extractor is not too high:

Claim 13. $k' = k + O(\log^3 \frac{n}{\varepsilon})$

Proof. Again, a simple computation shows that for some constants c_1, c_2, c_3:

$$\begin{aligned}
k' &= z_0 + w + \left(\sum_{i=1}^{\ell} z_i + w\right) \cdot 2^{\ell} \\
&= k + w + \left(\sum_{i=1}^{\ell} z'_i + \ell\left(2\log\frac{1}{\varepsilon} + \log 4|F| + \log 2\ell\right) + w\right) \cdot 2^{\ell} \\
&\leq k + w + \left(\sum_{i=1}^{\ell} z'_i + c_1\left(\log\frac{n}{\varepsilon} + \log\log\log\frac{n}{\varepsilon}\right)\right) \cdot 2^{\ell} \\
&= k + w + \left(\sum_{i=1}^{\ell} 2^i \cdot c\log\frac{n}{\varepsilon} + c_1\left(\log\frac{n}{\varepsilon} + \log\log\log\frac{n}{\varepsilon}\right)\right) \cdot 2 \cdot C_{guv}\log\frac{n}{\varepsilon} \\
&\leq k + w + \left(c_2 \cdot c\log^2\frac{n}{\varepsilon} + c_1\left(\log\frac{n}{\varepsilon} + \log\log\log\frac{n}{\varepsilon}\right)\right) \cdot 2 \cdot C_{guv}\log\frac{n}{\varepsilon} \\
&\leq k + c_3\log^3\frac{n}{\varepsilon}
\end{aligned}$$

□

Together with the block wise extractor E, the segmentation family F and Claim 13, by Theorem 5 we can construct a (k', ε)-somewhere random extractor as needed

□

D A simple construction with polylogarithmic entropy loss

In this section we will show a simple construction due to Ronen Shaltiel for dispersers with polylogarithmic entropy loss.

Theorem 9. *There exists a constant c such that for any $\varepsilon > 0$ and $c\log\frac{n}{\varepsilon} < k \leq n$ there exists a (k_1, ε)-somewhere random extractor*

$$S : \{0,1\}^n \times \{0,1\}^d \to \left(\{0,1\}^m\right)^{O(k_1)}$$

where $k_1 = k + O\left(\log k \cdot \log\frac{k}{\varepsilon}\right), d = O\left(\log\frac{n}{\varepsilon}\right), m = k + O\left(\log k \cdot \log\frac{k}{\varepsilon}\right)$. Furthermore, the entropy loss of S is $\Delta(S) = O\left(\log k \cdot \log\frac{k}{\varepsilon}\right)$

Proof. The technique we use is due to [NTS99]. The idea behind it is that given two (k_1, ε_1) and (k_2, ε_2)-extractors E_1, E_2 and a $(k_1 + k_2 + s)$-source X (where s is some extra "safety" bits of entropy) there is a segmentation of X into two parts $X_{[1,i]}, X_{[i+1,n]}$ such that $X_{[1,i]}$ is a k_1 source and $X_{[i+1,n]}$ is a k_2 source. With this, by corollary 2, $E_2(X_{[1,i]}, E_1(X_{[i+1,n]}, U))$ is roughly $(\varepsilon_1 + \varepsilon_2)$-away from uniform. Since we don't know the precise value of i, we run our composition on all is and get a somewhere random extractor with n output blocks. The process is made precise in the following composition lemma:

Lemma 14. *(Theorem 3 in [NTS99]) Let* $E_1 : \{0,1\}^n \times \{0,1\}^{t_1} \to \{0,1\}^{t_2}$ *and* $E_2 : \{0,1\}^n \times \{0,1\}^{t_2} \to \{0,1\}^m$ *be (respectively)* (k_1, ε_1) *and* (k_2, ε_2)*-extractors, then for any* $s > 0$ *there exists a* $(k_1 + k_2 + s, \varepsilon_1 + \varepsilon_2 + 8n \cdot 2^{-s/3})$*-somewhere random extractor*

$$S : \{0,1\}^n \times \{0,1\}^{t_1} \to (\{0,1\}^m)^n$$

Let $\varepsilon' = \varepsilon/4$ and let X be a k_1-source where $k_1 = k + c_1 \log k \cdot \log \frac{k}{\varepsilon'} + s$ for some sufficiently large constant c_1 and s to be specified later. We will first condense our source and then apply the lemma on two extractors we present shortly.

D.1 The condenser

We will use the condenser of theorem 8. Let

$$C : \{0,1\}^n \times \{0,1\}^{d_1} \to \{0,1\}^{2(d_1+k_1)}$$

be a $k_1 \to_{\varepsilon'} k_1 + d_1$ condenser where $d_1 = O(\log \frac{n}{\varepsilon'})$. We again pick $k \geq c^* \log \frac{n}{\varepsilon'}$ for some constant c^* such that $k_1 \geq d_1$.

D.2 The extractors

We will need two extractors. The first is given by the following theorem:

Theorem 10. *(Theorem 4.19 in [GUV09] with* $\alpha = 1/2$*) There exists a constant* c *such that for all positive integers* $n \geq k$ *and all* $\varepsilon > 0$*, there is an explicit* (k, ε)*-extractor* $E_{GUV} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *where* $m = k/2$ *and* $d = \log n + c \log(\frac{k}{\varepsilon})$*.*

With the above, let

$$E_1 : \{0,1\}^{2(d_1+k_1)} \times \{0,1\}^{d_2} \to \{0,1\}^{\frac{c_1}{2}\log k \cdot \log \frac{k}{\varepsilon'}}$$

be a $(c_1 \log k \cdot \log \frac{k}{\varepsilon'}, \varepsilon')$-extractor where $d_2 = \log 2(d_1 + k_1) + c \log \left(\frac{c_1 \log k \cdot \log \frac{k}{\varepsilon'}}{\varepsilon'} \right) = O(\log \frac{k}{\varepsilon'})$.

Our second extractor will use theorem 7 again. Let

$$E_2 : \{0,1\}^{2(d_1+k_1)} \times \{0,1\}^{d_3} \to \{0,1\}^m$$

be a (k, ε')-extractor where $d_3 = \log 2(d_1 + k_1) + C_{guv} \log k \log \frac{k}{\varepsilon'}$ and $m = k + d_3 - 2 \log \frac{1}{\varepsilon'} - O(1)$.

D.3 The composition

Given a k_1 source X, let $Z = C(X, U_{d_1})$ be the output of the condenser. We let $c_1 = 2C_{guv}$ and our somewhere random extractor will output:

$$S(X, U_{d_1} \circ U_{d_2}) = E_2(Z, E_1(Z, U_{d_2}))$$

Since Z is ε'-close to a $(d_1 + k_1)$-source it follows by corollary 2 and lemma 14 that S is a $(k_1, 3\varepsilon' + 16(d_1 + k_1) \cdot 2^{-s/3})$. As $d_1 + k_1 \leq 2k_1$ if we pick $s = 3 \cdot \log \frac{32k_1}{\varepsilon'}$ we get that

$$3\varepsilon' + 16(d_1 + k_1) \cdot 2^{-s/3} \leq 4\varepsilon' = \varepsilon$$

and thus S is a (k_1, ε)-somewhere random extractor.

D.4 The parameters

We require $k_1 = k + c_1 \log k \cdot \log \frac{k}{\varepsilon'} + s$ bits of entropy for our source, d_1 bits of entropy for the uniform seed of C and d_2 bits of entropy for the uniform seed of E_1. We first note that indeed $d_1 + d_2 = O(\log \frac{n}{\varepsilon})$. Next, S has $2(d_1 + k_1) \leq 4k_1$ output blocks and finally, for the entropy loss, the output of the somewhere random extractor has length $k + d_3 - 2\log\frac{1}{\varepsilon'} - O(1) = k + \log 2(d_1 + k_1) + \frac{c_1}{2}\log k \log \frac{k}{\varepsilon'} - 2\log\frac{1}{\varepsilon} - O(1)$ and thus the loss is:

$$k + c_1 \log k \cdot \log \frac{k}{\varepsilon'} + d_1 + d_2 - (k + \log 2(d_1 + k_1) + \frac{c_1}{2} \log k \log \frac{k}{\varepsilon'} - 2\log\frac{1}{\varepsilon} - O(1))$$

We see that apart from the loss of $\frac{c_1}{2}\log k \log \frac{k}{\varepsilon'}$ all other factors in the loss are of order $O(\log \frac{k}{\varepsilon})$ and we are done. □