03683170: Expanders, Pseudorandomness and Derandomization 15/05/16

Lecture 5 – List-decoding, PRGs and Extractors

Amnon Ta-Shma and Dean Doron

1 Hashing, mixing, encoding, correcting, PRGs and more...

We encounter hash functions all the time. Often we want to hash a small set in a large universe into a smaller domain in a (sometimes almost) one-to-one way. Sometimes we do not know the small set (but know that it is small) and still want the hashing to work. To achieve this, we need a probabilistic algorithm and we need to accept an unavoidable, but small, probability of collision.

Sometimes, we hash the other way around. That is, we would like to start with a small set and hash it into a larger universe so that the elements in the image are dispersed. A popular choice is using the Hamming distance and then we get an error correcting code that guarantees that all elements in the image (i.e., all codewords) are far apart. This in particular allows error correction up to half the distance.

In the extraction problem we are given a small unknown set and we would like to extract randomness from it. Let us concentrate on just outputting a single additional bit, which is the simplest task we can ask for, yet already captures the conceptual essence of the problem. We would like to somehow take an input from an unknown source and mix all the bits together so that wherever the entropy was, the bits are now mixed and every bit we choose is (marginally) close to uniform.

Trevisan [7] made the striking observation that an error correcting code does exactly that. Intuitively, a single change in one bit between the inputs should result in two codewords that are different all over (and maximally different when the relative distance is close to half), which in particular implies that every output bit in the code essentially depends on almost half of the input bits. Thus, in a sense, an error correcting bits mixes bits well!

The exact notion capturing this intuition is that of *list decoding*. An $[\bar{n}, n, \delta]_q$ error correcting code C is (L, ζ) list-decodable if for every word $w \in \mathbb{F}_q^{\bar{n}}$, the number of codewords of C in a ball of radius $\zeta \cdot \bar{n}$ around w is at most L.

The notion of list-decoding was proposed by Elias [1]. If we use a binary code with non-negligible rate then its relative distance is strictly less than half and therefore we can uniquely decode less than a quarter fraction of errors. With list decoding we can approach half! Similarly, for a larger alphabet size, however large q is, we can never correct half errors, but with list-decoding we can approach 1.

Of course, with list decoding the answer is often not unique. With stochastic models of error we can almost always suffer closer and closer to the distance errors (and this was first proved by Shannon in his famous paper). But in the adversarial model we use, the adversary can always choose w half way between two codewords. Can there be much more? The question of whether list-decodable codes are useful, mainly depend on how small we can guarantee L is.

A good code with a good distance has remarkable list-decoding capabilities. This is captured by the Johnson's bound, with which we start. Some codes (and, in particular, random codes) do much better, and there is a beautiful chain of papers leading to remarkable explicit codes with better

(almost optimal) list-decoding. These works turned out also to be closely connected to explicit, almost optimal extractors, but we will not explore this path in the course (we cover it in the seminar, and you are all very welcome to join).

As we said before, Trevisan proved an equivalence between strong seeded extractors with just a single output bit and list-decodable codes, and we will see that next. Trevisan then continued (in the same paper!) to show that the NW generator is actually an extractor. An appealing property of both constructions (strong extractor with a single output bit, and the extractor emerging from the NW generator) is that they come equipped with a *reconstruction algorithm*, and we conclude this chapter with the definition of a reconstruction extractor and its connection to PRG, and prove Trevisan's observation as a corollary.

2 List-decodable codes and the Johnson bound

A good place to read about the material in this section is Chapter 7 of the (online) book by Guruswami, Rudra and Sudan [2]. Section 7.2 defines the notion of list-decoding. Section 7.3 is the Johnson bound. Section 7.4 discusses the optimal list-decoding capacity. In class we will cover only a subset of the material in the book and we encourage you to read the entire chapter to complete the picture. Below we collect what we need for later on.

Definition 1. *An $[\bar{n}, n, \delta]_q$ error correcting code C is (L, ζ) list-decodable if for every word $w \in \mathbb{F}_q^{\bar{n}}$, the number of codewords of C in a ball of radius $\zeta \cdot \bar{n}$ around w is at most L.*

What can we hope for?

Over alphabet size q, we cannot hope to correct a fraction of over $1 - \frac{1}{q}$ errors, because total noise wipes all information while still leaving (w.h.p.) $\frac{1}{q}$ fraction of agreement with the original, unspoiled codeword. Indeed, we can get as close as we want to $\frac{1}{q}$ agreement, but we pay for that in the rate, and if we insist on being very close to $1 - \frac{1}{q}$ we pay for this also in a larger and larger list size. Formally,

Theorem 2. *Let $q \geq 2$, $0 \leq p < 1 - \frac{1}{q}$, and $\varepsilon > 0$. The following holds for large enough n:*

1. *If $R \leq 1 - H_q(p) - \varepsilon$, then there exists a $(p, O(\varepsilon^{-1}))$ list-decodable code with rate R,*
2. *If $R > 1 - H_q(p) + \varepsilon$, every (p, L) list-decodable code with rate R has $L \geq q^{\Omega(\varepsilon \bar{n})}$,*

and $H_q(x) = x \log_q(q-1) - x \log_q x - (1-x) \log_q(1-x)$ is the entropy function.

Next, we show that distance implies decent list-decoding.

Theorem 3. *If $e \leq \bar{n} - \sqrt{\bar{n}(\bar{n} - d)}$ then any $[\bar{n}, n, d]_q$ code is $(e/\bar{n}, q\bar{n}d)$ list-decodable, for all q.*

If we want to be more precise (and also give a more accurate upper bound on the list size, that is often just a constant) we have:

Theorem 4. *Let C be a $[\bar{n}, n, d]_q$ code. If $\zeta < J_q(d/\bar{n})$ then C is $(\zeta, qd\bar{n})$ list-decodable, where*

$$J_q(x) \;=\; \left(1 - \frac{1}{q}\right)\left(1 - \sqrt{1 - \frac{qx}{q-1}}\right).$$

Also, $J_q(x)$ satisfies $J_q(x) \geq 1 - \sqrt{1-x} \geq x/2$ for $0 \leq x \leq 1 - \frac{1}{q}$.

We shall also use the following explicit list-decoding result:

Lemma 5. *For every n and δ, there is an efficient $[\bar{n}, n, \frac{1}{2}-\delta]_2$ code with $\bar{n} = \mathrm{poly}(n, \delta^{-1})$ that is $(\frac{1}{2}-\delta, \delta^{-2})$ list-decodable.*

3 The equivalence between strong extractors with one output bit and list-decodable error-correcting codes

3.1 Strong extractors imply list-decodable codes

Let $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}$ be a strong (k, ε) extractor and identify $\{0,1\}^d$ with $\{1, \ldots, D\}$. Define the code

$$C_E \;=\; \{(E(x,1), \ldots, E(x,D)) \mid x \in \{0,1\}^n\}$$

We prove:

Theorem 6. *C_E is $(2^k - 1, \frac{1}{2} - \varepsilon)$ list-decodable.*

Proof. Let $w \in \{0,1\}^D$ and let $\mathcal{A} = B(w, \varepsilon)$ denote the set of all codewords of C_E in a ball of radius $\left(\frac{1}{2} - \varepsilon\right) \cdot \bar{n}$ around w. Further, define $T : [D] \times \{0,1\} \to \{0,1\}$ such that $T(y,b)$ returns 1 iff $w(y) = b$.

For $\mathcal{X} \subseteq \{0,1\}^n$ denote $P_{E,\mathcal{X}} = Y \circ E(X,Y)$ where X is the flat distribution over $\mathcal{X}$ and Y is the uniform distribution over $[D]$. Consider the distribution $P_{E,\mathcal{A}}$. It holds that

$$\Pr[T(P_{E,\mathcal{A}}) = 1] = \frac{1}{|\mathcal{A}|} \sum_{\bar{x} \in \mathcal{A}} \Pr_{y \sim U_d}[w(y) = E(\bar{x}, y)] > \frac{1}{|\mathcal{A}|} \sum_{\bar{x} \in \mathcal{A}} \left(\frac{1}{2} + \varepsilon\right) = \frac{1}{2} + \varepsilon.$$

Thus, $|P_{E,\mathcal{A}} - U_1| > \varepsilon$. As E is a (k, ε) strong extractor, it must be the case that $|\mathcal{A}| < 2^k$. □

One can easily see that this also holds for the non-binary case. That is, if $E : \mathbb{F}_q^n \times \{0,1\}^d \to \mathbb{F}_q$ is a strong (k, ε) extractor then C_E is $(2^k - 1, 1 - \frac{1}{q} - \varepsilon)$ list-decodable.

From the above theorem we can infer a lower-bound on the seed length of strong extractors. For simplicity, we shall give a relaxed bound.

Theorem 7. *Let $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}$ be a (k, ε) strong seeded extractor where $k \leq c\varepsilon^2 n$ for a large enough constant c. Then, $d \geq \log n + 2\log\frac{1}{\varepsilon} - 2$.*

Proof. Assume towards contradiction that $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}$ is a (k, ε) strong seeded extractor with $d \leq \log(n-k) + 2\log\frac{1}{\varepsilon} - 3$. Thus, C_E is $(2^k - 1, \frac{1}{2} - \varepsilon)$ list-decodable, $[D = 2^d, n, \cdot]_2$ error-correcting code. The rate of C is

$$R \;=\; \frac{n}{2^d} \;>\; 9\varepsilon^2.$$

We use the fact that $H_2(p) \geq 1 - 4(p - \frac{1}{2})^2$, so $H_2\left(\frac{1}{2} - \varepsilon\right) \geq 1 - 4\varepsilon^2$ and

$$R \;>\; 1 - H_2\left(\frac{1}{2} - \varepsilon\right) + 5\varepsilon^2.$$

By item (2) of Theorem 2, $L \geq 2^{c\varepsilon^2 n}$ for some constant c. By our restriction on k, $2^{c'\varepsilon^2 n} \geq 2^k$ and we have a contradiction. □

3.2 List decodable codes imply strong extractors

Let C be an $(\frac{1}{2}-\varepsilon, L)$ list-decodable $[\bar{n}, n, \cdot]_2$ code. Define $E_C : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}$ for $d = \log \bar{n}$ by $E_C(x,y) = C(x)_y$. We prove:

Theorem 8. *E_C is a strong $(k, 2\varepsilon)$ extractor for $k = \log \frac{L}{\varepsilon} + 1$.*

Proof. Assume towards contradiction that E_C is not a strong $(k, 2\varepsilon)$-extractor, so there exists an (n,k)-source X, so $|\mathrm{Supp}(X)| \geq \frac{2L}{\varepsilon}$, and a test $T : [\bar{n}] \times \{0,1\} \to \{0,1\}$ such that

$$\Pr_{x\sim X, y\sim U_d}[T(y, E(x,y)) = 1] - \Pr_{y\sim U_d, b\sim U_1}[T(y,b) = 1] > 2\varepsilon.$$

Hence, as we shall soon see, there exists a next-bit predictor $p : \{0,1\}^d \to \{0,1\}$ such that

$$\Pr_{x\sim X, y\sim U_d}[p(y) = E(x,y)] > \frac{1}{2} + 2\varepsilon.$$

By an averaging argument, there exists a set $G \subseteq \{0,1\}^n$ of weight at least ε such that for every $x \in G$, $\Pr_{y\sim U_d}[p(y) = E(x,y)] > \frac{1}{2} + \varepsilon$.

Now, let $z \in \{0,1\}^{\bar{n}}$ so that $z_i = p(i)$. For every $x \in G$, the Hamming distance between $C(x)$ and z is greater than $\left(\frac{1}{2} + \varepsilon\right)\bar{n}$. Hence, $C(x) \in B(z, \varepsilon)$ for every $x \in G$ so $\varepsilon \cdot |\mathrm{Supp}(X)| \leq L$, in contradiction. □

A similar claim can be made for non-binary codes and extractors. Let C be a $[\bar{n}, n, (1 - \frac{1}{q} - \varepsilon)\bar{n}]_q$ error-correcting code. Again, define $E_C : \mathbb{F}_q^n \times \{0,1\}^d \to \mathbb{F}_q$ for $d = \log \bar{n}$ by $E_C(x,y) = C(x)_y$.

Theorem 9 ([6])**.** *E_C is a strong $\left(\frac{1}{\varepsilon}, \sqrt{\frac{\varepsilon q}{2}}\right)$ extractor.*

Proof. The proof is very similar to the Leftover Hash Lemma we proved in Lecture 3. Let $\mathcal{X} \subseteq \mathbb{F}_q^n$ and recall that $P_{E,\mathcal{X}} = Y \circ E(X,Y)$ where X is the flat distribution over $\mathcal{X}$ and Y is the uniform distribution over $[\bar{n}]$. For $\mathcal{X}$ of cardinality at least $\frac{1}{\varepsilon}$, let us compute the collision probability of $P_{E,\mathcal{X}}$:

$$\begin{aligned}
\| P_{E,\mathcal{X}} \|_2^2 &= \Pr_{y_1,y_2\sim U_d}[y_1 = y_2] \cdot \left(\Pr_{x_1,x_2\sim X}[x_1 = x_2] + \Pr_{x_1,x_2\sim X, y\sim U_d}[E(x_1,y) = E(x_2,y) \mid x_1 \neq x_2] \right) \\
&= \frac{1}{\bar{n}} \cdot \left(\frac{1}{|\mathcal{X}|} + \Pr_{x_1,x_2\sim X, y\sim U_d}[C(x_1)_y = C(x_2)_y \mid x_1 \neq x_2] \right) \\
&\leq \frac{1}{\bar{n}} \cdot \left(\frac{1}{|\mathcal{X}|} + \frac{1}{q} + \varepsilon \right) = \frac{1}{q\bar{n}} \cdot \left(1 + q\left(\frac{1}{|\mathcal{X}|} + \varepsilon \right) \right) \leq \frac{1}{q\bar{n}}(1 + 2\varepsilon q).
\end{aligned}$$

We make use of the following claim:

Claim 10. *If X is a distribution such that $\| X \|_2^2 \leq \frac{1}{|\mathrm{Supp}(X)|}(1 + 4\delta^2)$ then X is δ-close to uniform.*

Now, $|\mathrm{Supp}(P_{E,\mathcal{X}})| \leq q\bar{n}$, so $P_{E,\mathcal{X}}$ is $\sqrt{\frac{\varepsilon q}{2}}$-close to uniform, as desired. □

From the above theorem we can re-prove the Johnson bound.

Theorem 11. *Let C be a $[\bar{n}, n, (1-\frac{1}{q}-\varepsilon)\bar{n}]_q$ error-correcting code. Then, it is $(1-\frac{1}{q}-\zeta, \frac{q}{4\zeta^2-q\varepsilon})$ list-decodable for every $\zeta > \frac{1}{2}\sqrt{\varepsilon q}$.*

Proof. By the above theorem, E_C is a strong $\left(\frac{1}{\varepsilon}, \sqrt{\frac{\varepsilon q}{2}}\right)$ extractor. Let $w \in \{0,1\}^{\bar{n}}$ and let $\mathcal{A} = B(w, 1-\frac{1}{q}-\zeta)$ for $\zeta > \frac{1}{2}\sqrt{\varepsilon q}$. Thus:

- On one hand, by inspection of Theorem 6 we see that $P_{E,\mathcal{A}}$ is ζ-far from the uniform distribution.
- On the other hand, by inspection of Theorem 9 we see that $P_{E,\mathcal{A}}$ is $\frac{1}{2}\sqrt{q\left(\frac{1}{|\mathcal{A}|}+\varepsilon\right)}$-close to the uniform distribution.

Overall, $|\mathcal{A}| \leq \frac{q}{4\zeta^2-\varepsilon q}$, as desired. □

4 Nisan's generator: A PRG against random AC^0 circuits

4.1 Weak Designs

Definition 12 (A design [4])**.** *A family of sets $Z_1, Z_2, \ldots, Z_m \subseteq [t]$ is a (ℓ, a) design if*

1. *For all $i \in [t]$, $|Z_i| = \ell$, and*
2. *For all $i \neq j$, $|Z_i \cap Z_j| \leq a$.*

Claim 13. *For every ℓ, m there exists a $(\ell, a = \log m)$ design $Z_1, \ldots, Z_m \subseteq [t]$ where $t = O(\ell^2)$.*

Proof. Assume w.l.o.g. that ℓ is a prime power. Consider the numbers in $[t]$ as pairs of elements in $\mathbb{F}_\ell$. I.e., identify $[t]$ with $\{(x,y) \mid x, y \in \mathbb{F}_\ell\}$.

For every polynomial $p \in \mathbb{F}_\ell[X]$ of degree at most a, define the set of all evaluations $S_p = \{(x, p(x)) \mid x \in \mathbb{F}_\ell\}$. There are at least $\ell^{a+1} \geq m$ such polynomials, so all that is left is to observe that:

1. For every p, $|S_p| = \ell$.
2. For every $p_1 \neq p_2$, $|S_{p_1} \cap S_{p_2}| \leq a$.

Therefore, every m sets from $\{S_p\}_p$ is a (ℓ, a) design. □

In fact, a slightly more refined notion that already suffices is of a weak design:

Definition 14 (Weak design [5])**.** *A family of sets $Z_1, \ldots, Z_m \subseteq [t]$ is a weak (ℓ, ρ) design if*

1. *For all $i \in [t]$, $|Z_i| = \ell$, and*
2. *For all $i \neq j$, $\sum_{j<i} 2^{|Z_i \cap Z_j|} \leq \rho \cdot (m-1)$.*

We cite without a proof:

Lemma 15 ([5])**.** *For every ℓ, m and $\rho > 1$, there exists a weak (ℓ, ρ) design $Z_1, \ldots Z_m \subseteq [t]$ with $t = \left\lceil \frac{\ell}{\ln \rho} \right\rceil \cdot \ell$. Such a family can be found in time* $\mathrm{poly}(m, t)$.

4.2 The NW-generator

We would like to construct a pseudo-random generator (PRG) fooling AC^0. A PRG against a class of functions F is a function $G : \{0,1\}^\ell \to \{0,1\}^n$ such that no function $f \in F$ ε-distinguishes $G(U_\ell)$ from the uniform distribution. Alternatively, one can view the uniform distribution over the set $G(\{0,1\}^\ell)$ as a distribution with a (very small) support that is a good replacement to the completely uniform distribution as long as the uniformity test is done by a function from the restricted class (AC^0 in our case).

Let $f : \{0,1\}^\ell \to \{0,1\}$ be a boolean function that is hard on average for AC^0 (although the argument that follows can be naturally be extended to every function f that is hard on-average for some class). Recall, for example, that Parity is such a function. Also, let $S_1, \ldots, S_m \subseteq [t]$ be a $(\ell, 2)$ weak design that is guaranteed by Lemma 15. The generator $G : \{0,1\}^t \to \{0,1\}^m$ is given by:

$$G(y) \;=\; f(y|_{S_1}), \ldots, f(y|_{S_m}).$$

We claim:

Lemma 16 ([4])**.** *If there exists an* AC *circuit* $D : \{0,1\}^m \to \{0,1\}$ *of size* s *and depth* d *such that* $\Pr_{x\sim U_t}[D(G(x)) = 1] > \Pr_{x\sim U_m}[D(x) = 1] + \varepsilon$ *then there exists a circuit* $D'' : \{0,1\}^\ell \to \{0,1\}$ *of size* $O(s)$ *and depth* $d+2$ *that computes* f *successfully with probability larger than* $\frac{1}{2} + \frac{\varepsilon}{m}$*.*

Proof. The proof begins by a hybrid argument (due to Yao [8]). As $\Pr_{x\sim U_t}[D(G(x)) = 1] - \Pr_{x\sim U_m}[D(x) = 1] > \varepsilon$, we can write

$$\varepsilon \;<\; \sum_{i\in[m]} \left(\Pr_{y\sim U_t, \bar{x}\sim U_{m-i}}[D(G(y)_{1,\ldots,i}, \bar{x}) = 1] - \Pr_{y\sim U_t, \bar{x}\sim U_{m-i+1}}[D(G(y)_{1,\ldots,i-1}, \bar{x}) = 1] \right)$$

and there exists $i \in [m]$ and a fixing of the uniform variables into D for which

$$\Pr_{y\sim U_t}[D(G(y)_{1,\ldots,i}) = 1] - \Pr_{y\sim U_t, x\sim U_1}[D(G(y)_{1,\ldots,i-1}, x) = 1] \;>\; \frac{\varepsilon}{m}.$$

Denote $y = (y_2, y_3)$ to indicate that $y_2 \in \{0,1\}^{t-\ell}$ are the bits outside S_i and $y_3 \in \{0,1\}^\ell$ are the bits in S_i. Hence, there exists a fixing of the bits outside S_i so that

$$\Pr_{y\sim \bar{Y}}[D(G(y)_{1,\ldots,i}) = 1] - \Pr_{y\sim \bar{Y}, x\sim U_1}[D(G(y)_{1,\ldots,i-1}, x) = 1] \;>\; \frac{\varepsilon}{m}.$$

where $\bar{Y}$ is uniform on S_i and fixed to y_2 elsewhere. Fixing y_2 into D, we have that

$$\Pr_{y_3\sim U_\ell}[D(f(y|_{S_1}), \ldots, f(y|_{S_{i-1}}), f(y_3)) = 1] - \Pr_{y_3\sim U_\ell, x\sim U_1}[D(f(y|_{S_1}), \ldots, f(y|_{S_{i-1}}), x) = 1] > \frac{\varepsilon}{m}.$$

Denote $D'(y_3, r) = D(f(y|_{S_1}), \ldots, f(y|_{S_{i-1}}), r)$, so we have

$$\Pr_{y_3\sim U_\ell}[D'(y_3, f(y_3)) = 1] - \Pr_{y_3\sim U_\ell, x\sim U_1}[D'(y_3, x) = 1] > \frac{\varepsilon}{m}.$$

The distinguishing circuit D' gives rise to a next-bit predictor circuit D'' such that

$$\Pr_{y_3\sim U_\ell, D''}[D''(y_3) = f(y_3)] > \frac{1}{2} + \frac{\varepsilon}{m}.$$

D'' on input y_3 picks a random bit b and computes $D'(y_3, b)$. If D' outputs 1 then D'' returns b and otherwise it returns $1 - b$. The randomness of D'' can be fixed. We will prove the correctness of this procedure in the exercise.

We see that $|D''| = |D'|$. What is the size of D'? We replace inputs of D with values of f on at most $\sum_{j<i} 2^{S_i \cap S_j} \leq \rho m$ different inputs. Thus, $|D''| = |D| + O(\rho m) = |D| + O(m) = O(|D|)$ and there is only a depth 2 increase in evaluating these truth tables. □

Recall that:

Theorem 17 ([3])**.** *Let C be a circuit of depth d and size $2^{O(\ell^{1/d})}$. Then,*

$$\Pr_{x \sim U_\ell}[C(x) = \text{Parity}(x)] \;\leq\; \frac{1}{2} + 2^{-\Omega(\ell^{1/d})}.$$

Combining this theorem with the above lemma, we obtain:

Corollary 18. *$G : \{0,1\}^t \to \{0,1\}^m$ with $t = 4\log^{4d} m$ is a PRG against AC^0 circuits of depth d with error $\varepsilon = \frac{1}{m}$.*

Proof. By Lemma 15 we can take $\ell = \frac{\sqrt{t}}{2} = \log^{2d} m$. Assume towards contradiction that G is not a PRG and let D be an AC circuit of size $s = \text{poly}(m)$ and depth d which G does not ε-fool. By Lemma 16 there exists a circuit of depth $d + 2$ and size $s' = \text{poly}(m)$ that computes Parity on ℓ bits with success probability larger than $\frac{1}{2} + \frac{1}{m^2}$.

By Theorem 17, the size of D is at least $2^{\log^{1.5} m}$ if $\frac{1}{m^2} > 2^{-\log^{1.5} m}$, which is indeed the case. However, the size of D is only polynomial in m, in contradiction. □

As another corollary, we have the following derandomization result:

Corollary 19. $\mathsf{RAC}^0 \subseteq \bigcup_c \mathsf{DSPACE}(\log^c n)$.

We leave its proof to the reader.

5 Reconstruction extractors and PRGs

Let us first omit details and parameters, and ignore issues of worst-case vs. average-case hardness, an then give a rigorous and formal treatment of this material.

An efficient PRG implies an explicit function in the complexity class E that is hard for small non-uniform circuits [4]. The converse is also true, but harder to prove. The first result of this kind is Nisan's generator for AC^0 that we have seen (and later generalized in the Nisan-Wigderson construction).

The NW construction and also the later improvements are black-box constructions in the following sense: They start with an explicit function $f : \{0,1\}^\ell \to \{0,1\}$ and construct from it a new function $G^f : \{0,1\}^t \to \{0,1\}^m$ (where the notation is meant to indicate that G makes black-box oracle calls to f) and prove that if f is hard, then G^f is a PRG.

Most importantly for us, this implication is proved by exhibiting a "reconstruction" algorithm. Namely, the proof describes an efficient "reconstruction" oracle Turing Machine R such that for

every boolean function[1] $f : \{0,1\}^\ell \to \{0,1\}$, if there is a small circuit C that ε-distinguishes $G^f(U_t)$, then there exists a *short* advice string $z = A(f)$ such that $R^C(z,i)$ computes $f(i)$. In particular the existence of R implies:

Lemma 20 (informal, [4])**.** *If $f : \{0,1\}^\ell \to \{0,1\}$ is suitably hard then G^f is a pseudorandom generator.*

Proof. (sketch) If there is a small circuit C that ε-distinguishes $G^f(U_t)$, then by hardwiring the "correct" advice $z = A(f)$, $R^C(z,i)$ is a small circuit computing f. The contrapositive then says that if f cannot be computed by small circuits, then $G^f(U_t)$ is a PRG. □

The above result is conditional: if f is a hard function then G^f is a PRG. Trevisan showed that reconstructive PRG are strong enough to give an *unconditional* extractor construction:

Lemma 21 (informal, [7])**.** *$E : \{0,1\}^{2^\ell} \times \{0,1\}^t \to \{0,1\}^m$ defined by $E(f,y) = G^f(y)$ is an extractor.*

Proof. (sketch) Let $n = 2^\ell$ and let $X \subseteq \{0,1\}^n$ be a large subset. We identify $\{0,1\}^n$ with the set of all functions from $\{0,1\}^\ell$ to $\{0,1\}$. If $E(X,U_t)$ is not close to uniform, then there exists a function C that ϵ-distinguishes $E(X,U_t)$. By averaging, we can say that C $\frac{\epsilon}{2}$-distinguishes $E(x,U_t)$, for many $x \in X$. Therefore, for many $x \in X$ there exists a short advice string $z = A(x)$ for which $R^C(z,\cdot)$ outputs x. The number of strings x with such short descriptions cannot exceed the number of possible advice strings. We conclude that if $E(X,U_t)$ is not close to uniform, then X is small. The contrapositive says that if X is large, then $E(X,U_t)$ is close to uniform; in other words, E is an extractor. □

We now want to make these notions formal. We define:

Definition 22 (reconstructive PRG)**.** *Let (G,A,R) be of functions where:*

- *$G : \{0,1\}^t \times \to \{0,1\}^m$ is called the generator function,*
- *$A : \{0,1\}^{2^\ell} \to \{0,1\}^a$ is called the advice function,*
- *$R : \{0,1\}^a \times \{0,1\}^\ell \to \{0,1\}$ is called the reconstruction function,*

such that $G = G^f$ and $R = R^C$ can be computed by oracle circuits*, circuits that can use gates for functions f and C, respectively. (G,A,R) is a* (p,q)-reconstructive PRG *if, for every $f : \{0,1\}^\ell \to \{0,1\}$ and every distinguisher $C : \{0,1\}^m \to \{0,1\}$ that distinguishes $G^f(U_t)$ from U_m with advantage p, we have*

$$Pr_{i \sim \{0,1\}^\ell}[R^C(A(f),i) = f(i)] \;\geq\; q.$$

The following two theorems formalize Lemma 20 and Lemma 21, respectively:

Theorem 23. *Let (G,A,R) be a (p,q)-reconstructive PRG. Denote the depth and size of the circuit computing R by $s_R = s_R(a,\ell)$ and $d_R = d_R(a,\ell)$, respectively. Let $f : \{0,1\}^\ell \to \{0,1\}$ be a function that cannot be computed on average with $1-q$ error, using circuits of size $s = s(\ell)$ and depth $d = d(\ell)$. Then G is a PRG against circuits C of size $s_C = \frac{s}{s_R}$ and depth $s_C = \frac{d}{d_R}$, with error p.*

[1] We treat a boolean function and its truth-table interchangeably.

Proof. Assume by way of contradiction that there exists some circuit C as above that distinguishes $G(U_t)$ from U_m with advantage p. Let $z = A(f)$. By Definition 22, the circuit $R(z, \cdot)$ computes f on average with $1 - q$ error. The size and depth of $R(z, \cdot)$ are at most $s_R \cdot s_C = s$ and $d_R \cdot d_C = d$, respectively - a contradiction. □

Theorem 24. *Let (G, A, R) be a (p, q) reconstructive PRG, and let $n = 2^\ell$. Let $E : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ be defined as $E(f, y) = G^f(y)$, where we identify the set $\{0,1\}^n$ with the set of functions $f : \{0,1\}^\ell \to \{0,1\}$. Then E is a (k, ε) extractor, where $k = a + \log \sum_{i=0}^{(1-q)n} \binom{n}{i} + \log \frac{1}{p} + 1$ and $\varepsilon = 2p$.*

Proof. Assume by way of contradiction that E is not a (k, ε) extractor. Therefore, there is some flat distribution X, with support of size 2^k in $\{0,1\}^n$, such $|E(X, U_t) - U_m| > \varepsilon$.

Let G be the set of elements $f \in Supp(X)$ such that $|E(f, U_t) - U_m| > \frac{\varepsilon}{2}$. By an averaging argument, we have that $|G| > \frac{\varepsilon}{2} \cdot |Supp(X)| = p \cdot 2^k$.

To see this in detail, consider that $\varepsilon < |E(X, U_t) - U_m| = |\mathbb{E}_{f \sim X}[E(f, U_t) - U_m]|$. By the triangle inequality, $|\mathbb{E}_{f \sim X}[E(f, U_t) - U_m]| \leq \mathbb{E}_{f \sim X}[|E(f, U_t) - U_m|]$. From the law of total expectation,

$$\begin{aligned} \mathbb{E}_{f \sim X}[|E(f, U_t) - U_m|] &= \mathbb{E}_{f \sim X}[|E(f, U_t) - U_m| | f \in G] \Pr_{f \sim X}[f \in G] + \mathbb{E}_{f \sim X}[|E(f, U_t) - U_m| | f \notin G] \Pr_{f \sim X}[f \notin G] \\ &\leq \Pr_{f \sim X}[f \in G] + \frac{\varepsilon}{2}. \end{aligned}$$

This implies that $\Pr_{f \sim X}[f \in G] \geq \frac{\varepsilon}{2}$. Since X is a flat distribution, we have that $\Pr_{f \sim X}[f \in G] = \frac{|G|}{|Supp(X)|} = \frac{|G|}{2^k}$, and this gives us the required bound on the cardinality of G.

Now, for each $f \in G$, it holds by Definition 22 that for some $z \in \{0,1\}^a$, $R(z, \cdot)$ computes f on average with $1 - q$ error. By the (binary) Hamming bound, for every given $z \in \{0,1\}^a$, there are at most $\sum_{i=0}^{(1-q)n} \binom{n}{i}$ different functions that can be computed by $R(z, \cdot)$ with $1 - q$ error. Since there are 2^a possible values for z, we have that $p \cdot 2^k \leq |G| \leq 2^a \cdot \sum_{i=0}^{(1-q)n} \binom{n}{i}$. This implies $k \leq a + \log \sum_{i=0}^{(1-q)n} \binom{n}{i} + \log \frac{1}{p}$, a contradiction. □

In order to analyze Trevisan's extractor, we now present a definition similar to Definition 22, but different. Informally, instead of reconstructing a random coordinate of a given $x \in \{0,1\}^n$, we now choose x randomly from a subset X, and want to reconstruct x entirely. Here we also allow the reconstruction process to use randomness.

Definition 25 (reconstructive extractor)**.** *A triple (E, A, R) of functions where:*

- $E : \{0,1\}^n \times \{0,1\}^{r_E} \to \{0,1\}^m$ *is called the extractor function,*
- $A : \{0,1\}^n \times \{0,1\}^{r_A} \to \{0,1\}^a$ *is called the advice function, and,*
- $R : \{0,1\}^a \times \{0,1\}^{r_A} \times \{0,1\}^{r_R} \to \{0,1\}^n$ *is called the reconstruction function*

is a (p,q) reconstructive extractor *if for every* $X \subseteq \{0,1\}^n$ *and every next-bit predictor* $T : [m] \times \{0,1\}^m \to \{0,1\}$ *for* $E(X, U_{r_E})$ *with advantage* p*, we have*

$$\Pr_{x\sim X, y, z}[R^T(A(x,y), y, z) = x] \;\geq\; q.$$

Theorem 26. *If* (E, A, R) *as above is a* (p,q) *reconstructive extractor then* E *is a* (k,ε)*-extractor for* $k = a + \log\frac{1}{q} + 1$ *and* $\varepsilon = 2m \cdot p$*.*

Proof. Let (E, A, R) be a (p,q) reconstructive extractor. Assume by contradiction that E is not a (k,ε)-extractor. Therefore, there is some flat distribution X, with support of size 2^k in $\{0,1\}^n$, such that $|E(X, U_{r_E}) - U_m| > \varepsilon$. As we have seen, this implies the existence of a next-bit predictor T with advantage $\frac{\varepsilon}{2m} = p$. By definition, this implies that $\Pr_{x\sim X, y, z}[R^T(A(x,y), y, z) = x] \geq q$. Therefore, there are some fixed $y_0 \in \{0,1\}^{r_A}$, $z_0 \in \{0,1\}^{r_R}$, such that $\Pr_{x\sim X}[R^T(A(x, y_0), y_0, z_0) = x] \geq q$. Since X is a flat distribution, this means there are at least $q \cdot |Supp(X)| = q \cdot 2^k$ elements $x \in \{0,1\}^n$ for which $R^T(A(x, y_0), y_0, z_0) = x$. Since there are only 2^a possible inputs for $R^T(A(\cdot, y_0), y_0, z_0)$, we have that $q \cdot 2^k \leq 2^a$, or equivalently $k \leq a + \log\frac{1}{q}$, a contradiction. □

5.1 Trevisan's extractor

We now present Trevisan's extractor [7] and analyze its correctness using the reconstruction paradigm.

Trevisan's extractor

Parameters : n, ε and and $\rho = 2$. Set $\delta = \frac{\varepsilon}{2m}$.

Binary code : Let $C : \{0,1\}^n \to \{0,1\}^{\bar{n}}$ be a $(\frac{1}{2} - \delta, \delta^{-2})$ list-decodable code guaranteed by Lemma 5, and denote $\hat{x} = C(x)$.

Weak Design : A weak (ℓ, ρ) design $Z_1, \ldots, Z_m \subseteq [t]$, with:

- $\ell = \log \bar{n} = O(\log \frac{n}{\varepsilon})$,
- $t = \ell\lceil\frac{\ell}{\ln \rho}\rceil = O(\log^2 \frac{n}{\varepsilon})$.

Input : $x \in \{0,1\}^n$, $y \in \{0,1\}^t$.

Output : $TR(x,y) = \hat{x}(y|_{Z_1}), \ldots, \hat{x}(y|_{Z_m})$.

Lemma 27. *There exists functions* A *and* R *as above such that* (TR, A, R) *is a* $(\frac{\varepsilon}{m}, \frac{\varepsilon^3}{8m^3})$ *reconstructive extractor.*

Proof. Let $X \subseteq \{0,1\}^n$ be such that there exists $i \in [m]$ and $T : \{0,1\}^{i-1} \to \{0,1\}$ so that

$$\Pr_{x\sim X, y\sim U_t, T}[T(TR(x,y)_{1,\ldots,i-1}) = TR(x,y)_i] \;\geq\; \frac{1}{2} + \frac{\varepsilon}{m}$$

and T uses r_T bits of randomness. For $y \in \{0,1\}^t$, we write $y = (y_2, y_3)$ to say that y is $y_3 \in \{0,1\}^\ell$ on Z_i and $y_2 \in \{0,1\}^{t-\ell}$ on $[y] \setminus Z_i$. By an averaging argument, we have that

$$\Pr_{x\sim X, y_2\sim U_{t-\ell}}\left[\;\Pr_{y_3\sim U_\ell, T}[T(TR(x,y)_{1,\ldots,i-1}) = TR(x,y)_i] \;\geq\; \frac{1}{2} + \frac{\varepsilon}{2m}\;\right] \;\geq\; \frac{\varepsilon}{2m}.$$

Thus, there exists a set $G \subseteq \{0,1\}^n \times \{0,1\}^{t-\ell}$ of weight at least $\frac{\varepsilon}{2m}$ such that for every $(x, y_2) \in G$,

$$\Pr_{y_3 \sim U_\ell, T}[T(TR(x,y)_{1,\ldots,i-1}) = TR(x,y)_i] \geq \frac{1}{2} + \frac{\varepsilon}{2m}.$$

We shall now describe $A(x, y_2)$. A contains i and the truth tables for the bits $1, \ldots, i-1$, as determined by x, i and y_2. Namely, for $j < i$, we output all possible values for $\hat{x}(y|_{Z_j})$ in the following manner: For every $v \in \{0,1\}^{|Z_i \cap Z_j|}$, we construct a t-bits string y that is y_2 on $[t] \setminus Z_i$, v on $Z_i \cap S_j$ and 0 elsewhere. We then project y onto Z_j and apply $\hat{x}$.

Given $i \in [m]$, for every $j < i$ there are $2^{|Z_i \cap Z_j|}$ rows in the truth table. As $\sum_{j<i} 2^{|Z_i \cap Z_j|} \leq 2m$ we have that a, the length of $A(x, y_2)$, is $2m + \log m$.

The reconstruction procedure $R : \{0,1\}^a \times \{0,1\}^{r_a} \times \{0,1\}^{r_A} \to \{0,1\}^n$ is thus given by:

The reconstruction procedure R

Input : $A(x, y_2) = i, \{TT(i,j)\}_{j<i}$ and $y_2 \in \{0,1\}^{t-\ell}$.

Random coins : A random string $\alpha \in \{0,1\}^{r_T}$ and $\beta \in \left[\frac{4m^2}{\varepsilon^2}\right]$.

Algorithm :

- For every $y_3 \in \{0,1\}^\ell$,
  - Use y_2, y_3 and $TT(i,1), \ldots, T(i,i-1)$ to compute $\hat{x}(y|_{Z_1}), \ldots, \hat{x}(y|_{Z_{i-1}})$.
  - Compute $\hat{x}(y|_{Z_i}) = \hat{x}(y_3)$ by applying T, using α for its randomness.
- Let $\tilde{x}$ be the reconstructed $\hat{x}$ of the previous step.
- Apply list-decoding on $\tilde{x}$ and obtain a list $\mathcal{L}$ of size $\frac{4m^2}{\varepsilon^2}$.
- Use β to choose an element from $\mathcal{L}$ uniformly at random and output it.

Let $r_R = r_T + \log \frac{4m^2}{\varepsilon^2}$. For $(x, y_2) \in G$ we have that

$$\Pr_{z \sim U_{r_R}}\left[R^T(A(x,y_2), y_2, z) = x\right] \;\geq\; \frac{\varepsilon^2}{4m^2}.$$

Overall,

$$\Pr_{x \sim X, y_2 \sim U_{t-\ell}}\left[\Pr_{z \sim U_{r_R}}\left[R^T(A(x,y_2), y_2, z) = x\right] \;\geq\; \frac{\varepsilon^2}{4m^2}\right] \;\geq\; \frac{\varepsilon}{2m}$$

and the correctness follows. By Theorem 26, we conclude: □

Corollary 28. *$TR : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ is a $(2m + O(\log \frac{m}{\varepsilon}), 2\varepsilon)$ extractor with seed-length $t = O(\log^2 \frac{n}{\varepsilon})$.*

References

[1] Peter Elias. *List decoding for noisy channels.* Citeseer, 1957.

[2] Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. *Essential Coding Theory*. 2015.

[3] Johan Håstad. Computational limitations of small-depth circuits. 1987.

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

[5] Ran Raz, Omer Reingold, and Salil Vadhan. Extracting all the randomness and reducing the error in trevisan's extractors. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 149–158. ACM, 1999.

[6] Amnon Ta-Shma and David Zuckerman. Extractor codes. *Information Theory, IEEE Transactions on*, 50(12):3015–3025, 2004.

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

[8] Andrew C Yao. Theory and application of trapdoor functions. In *Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on*, pages 80–91. IEEE, 1982.