

Lecture 14: Extractors vs. Expanders

November 2, 2004

Scribe: Gholamreza Haffari

### 1 Existence of Extractors

Any  $\text{Ext} : \{0, 1\}^n \times \{0, 1\}^d \rightarrow \{0, 1\}^m$  can be viewed as a bipartite graph on  $N = 2^n$  vertices on left (which are labeled by samples of source  $X$ ),  $M = 2^m$  vertices on right (which are labeled by possible output strings), and with degree  $D = 2^d$  for the left vertices .

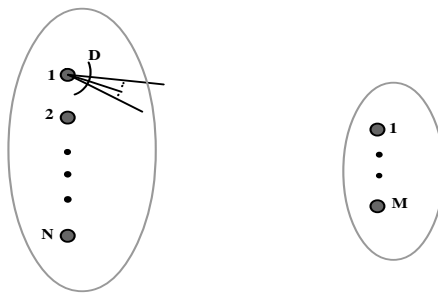


Figure 1: An extractor viewed as a bipartite graph

A vertex  $u$  on the left is connected to  $v$  on the right, if  $\text{Ext}(u, d) = v$  for some seed  $d$ .

A  $(k, \epsilon)$ -extractor on a flat  $k$ -source gives a distribution which is  $\epsilon$ -close to uniform. In particular, at most  $\epsilon$  fraction of vertices on the right side, may not have any incoming edges. In other words, for any set on the left side of size at list  $K$ , we will have  $|N(K)| \geq (1 - \epsilon)M$ . This suggests that a good extractor will be a good expander. But regarding the formal definition of expanders, we encounter two problems:

- The degree of extractors is not constant.
- What about expansion on large sets?

### 2 Extractors $\longrightarrow$ Expanders

We saw that an  $\text{Ext} : \{0, 1\}^n \times \{0, 1\}^d \rightarrow \{0, 1\}^m$  yields a bipartite graph  $G(N, M, E)$  of left degree  $D$ . Usually  $m \ll n$ , so our graph is unbalanced. (Look at the above figure.)

**Properties of  $G$ :**

Assume  $\text{Ext}$  is a  $(k, \epsilon)$ -extractor. For any set  $S \subset [N]$  of size at list  $K$  we have:

$$\Delta(\text{Ext}(S, U_d), U_m) \leq \epsilon$$

where  $S$  is a flat  $k$ -source, and  $\Delta$  denotes the statistical distance. This is equivalent to,  $\forall I \subseteq [M]$ ,

$$\left| \Pr[\text{Ext}(s, u_d) \in T] - \mu(T) \right| \leq \epsilon$$

Choose a random node from  $S$ . Then choose a random edge and check if the other side of this edge is in  $T$ .

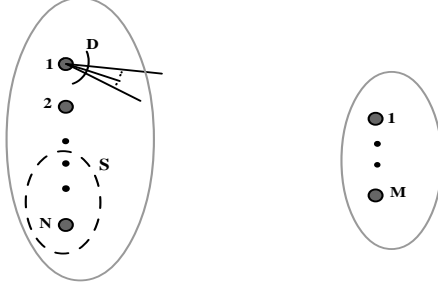


Figure 2:  $S \subset [N]$  of size at least  $K$ .

So,

$$\left| \frac{e(S, T)}{|S|D} - \mu(T) \right| \leq \epsilon$$

This inequality is similar to the one we had for expanders.

$$\left| \frac{e(S, T)}{ND} - \mu(S)\mu(T) \right| \leq \epsilon\mu(S)$$

Recall the Expander Mixing Lemma: For any  $\lambda$ -expander of degree  $D$  on  $N$  vertices, and for any sets  $S, T \subseteq [N]$ ,

$$\left| \frac{e(S, T)}{ND} - \mu(S)\mu(T) \right| \leq \lambda\sqrt{\mu(S)\mu(T)}$$

Comparing the two last inequalities we see that, to make an expander we need to guarantee  $\forall S, |S| \geq K$  and  $\forall T, \mu(T) \leq \frac{1}{2}$ :

$$\lambda\sqrt{\mu(S)\mu(T)} \leq \epsilon\mu(S)$$

So it suffices to have:

$$\lambda \leq \epsilon\sqrt{\frac{2K}{N}} \leq \epsilon\sqrt{\frac{\mu(S)}{\mu(T)}}$$

### 3 Expanders $\longrightarrow$ Extractors

Start with a  $D_0$ -regular  $\lambda_0$ -expander  $G_0$  on  $N$  vertices. Define  $G = G_0^t$  for some  $t$  such that  $\lambda_0^t \leq \epsilon\sqrt{\frac{2K}{N}}$ . So:

$$t = O\left(\log \frac{1}{\epsilon} + n - k\right)$$

$$D = D_0^t$$

This implies that the length of seed for our extractor is:

$$\log(D) = t \log(D_0) = O(n - k + \log \frac{1}{\epsilon})$$

To construct the extractor, put one copy of  $G$  as the left side, and one other copy as the right side. Use the same labeling for both sides:

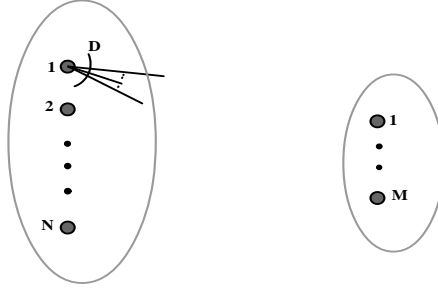


Figure 3: A left  $D$ -regular expander

Every vertex  $u$  on left side, should be connected to the vertices on the right which have labels, the same as neighbors of  $u$  in  $G$ . This way, we obtained a  $(k, \epsilon)$ -extractor  $\text{Ext} : \{0, 1\}^n \times \{0, 1\}^d \rightarrow \{0, 1\}^n$  with parameters:

$$n = \log N$$

$$d = O(n - k + \log \frac{1}{\epsilon})$$

Notice that the length of output is  $n$  bits ( $m = n$ )! Using this extractor, we can start with a weak source which has entropy  $k$ , and add  $O(n - k + \log \frac{1}{\epsilon})$  entropy in the form of seed, to get a random source with entropy  $n$ .

**Remark:**

- (1) For large  $k$ , (say  $k = n - \Delta$  for  $\Delta = O(\log n)$ ) the seed size is:  $d = O(\Delta + \log \frac{1}{\epsilon}) = O(\log n + \log \frac{1}{\epsilon})$  which is "small" compared to  $n$ .
- (2) The seed size is much better here than in the hash-based extractors where it was  $O(n)$ .

Recall the parameters of an *optimal*  $(k, \epsilon)$ -extractor  $\text{Ext} : \{0, 1\}^n \times \{0, 1\}^d \rightarrow \{0, 1\}^m$ :

$$m = k + d - 2 \log \frac{1}{\epsilon} - O(1)$$

$$d = \log(n - k) + 2 \log \frac{1}{\epsilon} + O(1)$$

For every set  $S \subseteq [N]$  of size  $K$  on the left side, and any set  $T \subseteq [M]$  on the right side, we have:

$$\left| \Pr[\text{Ext}(s, u_d) \in T] - \mu(T) \right| \leq \epsilon$$

If we take  $T \subseteq [M]$  to be  $\{a | \text{Ext}(s, u_d) \neq a\}$  we get:

$$\Pr[\text{Ext}(s, u_d) \in T] = 0$$

so for this  $T$ , we have  $\mu(T) \leq \epsilon$ . This means that for any set  $S \subseteq [N]$  of size at list  $K$ ,

$$|N(S)| \geq (1 - \epsilon)M$$

What "expanded" here, is the *density* of  $S$ , not its size! To calculate the expansion factor, assume  $|S| = K$ , then:

$$\frac{|N(S)|}{|S|} = \frac{|N(S)|}{K} \geq \frac{(1 - \epsilon)M}{K}$$

Suppose that the parameters are optimal. Then,

$$M = \Theta(KD\epsilon^2)$$

Hence:

$$\frac{|N(S)|}{|S|} \geq \frac{(1 - \epsilon)\Theta(KD\epsilon^2)}{K} \approx \epsilon^2 D$$

So the expansion factor we get in this way (which is  $\epsilon^2 D$ ), is not so much good compared to what we got in previous methods. (This was  $\frac{D}{2}$  for Ramanujan graphs.) The problem here is that we have put a too strong condition on extractors. They should act on a distribution with some entropy on left and give a distribution on the right side which is *close to uniform*. We may relax this condition and instead, assume the output distribution has some entropy or randomness. This change in definition will lead us to *conductors*. Then, the expander we construct using one conductor, will not have such a limitation on its expansion factor. Actually, we may combine conductors to make better ones. (And this is where the idea of zig-zag product comes from.)

The following table shows the differences between typical extractors and expanders.

<b>Expander</b> $G(N, \epsilon)$ <b>of degree</b> $D$	<b>Ext:</b> $[N] \times [D] \rightarrow [M]$
$D = \text{constant}$	$D = \text{polylog } n$
all sets of size $\leq K$ expand	all sets of size $\geq K$ expand
vertex/spectral expansion	min-entropy $\rightarrow$ stat. closeness to uniform distr.

## 4 Extraction from Block Sources

**Idea:** For simplicity consider the hash-based extractor we made:

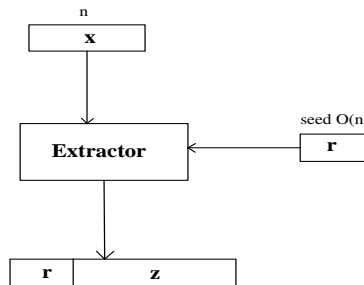


Figure 4: A hash-based extractor

We want to make this extractor better. If we assume that there is a random variables  $X'$  independent from  $X$ , then  $Z'$  and  $Z$  will be independent. So the output  $r.z.z'$  will be  $(2\epsilon)$ -close to uniform.

Now our idea is to split  $X$  to two (in some sense) "independent" parts and then, use the seed  $r$  to extract randomness from each part separately. At the end, concatenate the outputs with  $r$ .

**Definition 1 (block source)**  $X$  is a  $(k, l)$ -block source if  $X = B_1 \dots B_t$  (concatenation), where  $B_i = l$  and  $\forall b_1 \dots b_i \in \{0, 1\}^l$ ,

$$(B_i | b_1 \dots b_i) \text{ is a } k\text{-source}$$

**Theorem 2** Given an explicit construction of a (strong)  $(k, \epsilon)$ -extractor  $\text{Ext} : \{0, 1\}^l \times \{0, 1\}^d \rightarrow \{0, 1\}^{d+m}$ , we can give for any  $t \geq 1$ , an explicit construction of  $\text{Ext}' : \{0, 1\}^{tl} \times \{0, 1\}^d \rightarrow \{0, 1\}^{d+tm}$  such that for all  $(k, l)$ -block source  $X$  with  $t$  blocks,  $\text{Ext}'(X, U_d)$  is  $t\epsilon$ -close to uniform  $U_{d+tm}$ .

In this theorem, by "strong" we mean that the output includes the seed. Notice that the error adds up to  $t\epsilon$  which makes sense because we use the extractor  $t$  times.

**Proof:** This figure shows the idea of our construction:

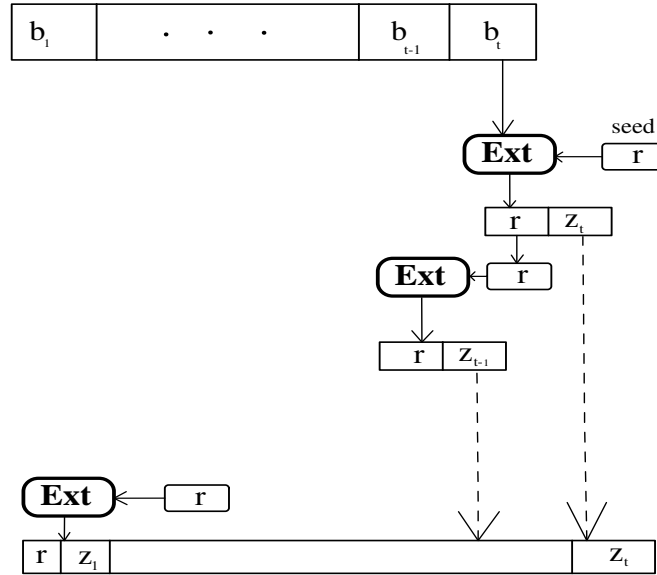


Figure 5: Extraction from block source

**Intuition:** By definition of extractors we know that  $\text{Ext}(B_t, U_d) \approx U_{d+m} = U_d \times U_m$ , so  $\text{Ext}(b_t, r) = (r, Z_t) \approx (r, u)$ . We can say that  $r$  is "approximately" independent of  $z_t$ . Hence, it can be used again to extract from  $b_{t-1}$

Case of two blocks: Let  $B_1$  and  $B_2$  be our blocks.  $R$  is the random variable corresponding to a seed  $r$ . Set,

$$(R, Z_2) = \text{Ext}(B_2, R)$$

$$(R, Z_1) = \text{Ext}(B_1, R)$$

we want to show that:  $(R, Z_1, Z_2) \stackrel{2\epsilon}{\approx} (R, M_1, M_2)$ , where  $M_1$  and  $M_2$  are independent copies of  $U_m$ . This can be easily seen applying triangle inequality to:

$$(R, Z_1, Z_2) = (\text{Ext}(B_1, R), Z_2) \stackrel{\epsilon}{\approx} (\text{Ext}(B_1, R), M_2) \stackrel{\epsilon}{\approx} (R, M_1, M_2)$$

The second approximating relation is the definition of extractor. We can concatenate  $(\text{Ext}(B_1, R)) \stackrel{\epsilon}{\approx} (R, M_1)$  with  $M_2$  because  $M_2$  is independent from others. For proving the first approximation look at:

$$\underbrace{(B_1, R, Z_2)}_{\text{Ext}} = (B_1, \text{Ext}(B_2, R)) \stackrel{\epsilon}{\approx} \underbrace{(B_1, R, M_2)}_{\text{Ext}}$$

Since Ext is a deterministic function it does not increase the distance and we can apply it to above to get:

$$(R, Z_1, Z_2) = (\text{Ext}(B_1, R), Z_2) \stackrel{\epsilon}{\approx} (\text{Ext}(B_1, R), M_2)$$

In general, for any  $t$ ,

$$\begin{aligned} & (B_1, \dots, B_{t-1}, \underbrace{B_t, R}_{\text{Ext}}) \\ & \stackrel{\epsilon}{\approx} (B_1, \dots, B_{t-1}, R_t, Z_t) \stackrel{\epsilon}{\approx} (B_1, \dots, \underbrace{B_{t-1}, U_d, M_t}_{\text{Ext}}) \\ \Rightarrow & (B_1, \dots, R_{t-1}, Z_{t-1}, Z_t) \stackrel{2\epsilon}{\approx} (B_1, \dots, \underbrace{B_{t-2}, U_d, M_{t-1}, M_t}_{\text{Ext}}) \\ & \vdots \\ \Rightarrow & (R_1, Z_1, \dots, Z_t) \stackrel{t\epsilon}{\approx} (M_1, \dots, M_t) \end{aligned}$$

■