CS225: Pseudorandomness Prof. Salil Vadhan

Lecture 13: More Connections with Expanders

March 22, 2007

Based on scribe notes by Adam Kirsch and Alexandr Andoni.

1 Lossless Condensers vs. Expanders

Last time we saw the notion of a $k \to_\varepsilon k'$ *condenser* $\mathrm{Con} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$, where for every k-source X on $\{0,1\}^n$, $\mathrm{Con}(X, U_d)$ is ε-close to some k'-source. We can define the *entropy loss* of a condenser to be $\ell = k' - (k+d)$. As we have discussed, an extractor (i.e. $m = k'$) must have $\ell \geq 2\log(1/\varepsilon) - O(1)$, whereas if we allow m to be larger than k' (specifically, $m \geq k' + \log(1/\varepsilon) + O(1)$), then it is possible for a condenser to be *lossless* (i.e. have $\ell = 0$).

As we have seen for extractors in Lecture Notes 11, every function $\mathrm{Con} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ can be viewed as a bipartite multigraph G with $N = 2^n$ left vertices, left-degree $D = 2^d$, and $M = 2^m$ right-vertices where the y'th neighbor of left-vertex x is $\mathrm{Con}(x,y)$. Generalizing what we showed for extractors, the condenser property implies a vertex-expansion property of the graph G. Specifically, if we define a $(= K, A)$ vertex expander to be one in which sets of size exactly K expand by a factor of A, then we have:

Lemma 1 *If* $\mathrm{Con} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is a* $k \to_\varepsilon k'$ *condenser, then the corresponding bipartite graph is a* $(= K, A)$ *vertex expander for* $A = (1-\varepsilon)\cdot K'/K = (1-\varepsilon)\cdot D/L$, *where* $K = 2^k$, $K' = 2^{k'}$, $D = 2^d$, $L = 2^\ell$, *and* $\ell = k' - (k+d)$ *is the entropy loss of the* Con.

Thus, if $L \leq D$, the expansion factor A is in fact bigger than 1 (but the case $A < 1$ is still interesting and nontrivial, because the graphs are unbalanced). In general, the vertex expansion property is weaker than the property of being a condenser — for example, when $m = k'$, it corresponds to a *disperser* rather than an extractor. However, for the special case of *lossless* condensers (i.e. $L = 1$), it turns out that the two properties are equivalent.

Lemma 2 *A function* $\mathrm{Con} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is a* $k \to_\varepsilon k+d$ *(lossless) condenser* if and only if *the corresponding bipartite graph is a* $(= K, A)$ *vertex expander for* $A = (1-\varepsilon)\cdot D$, *where* $K = 2^k \in \mathbb{N}$, $K' = 2^{k'}$, *and* $D = 2^d$.

Proof: The 'only if' direction follows from Lemma 1, so we only prove the 'if' direction. Assume that the bipartite graph corresponding to Con is a $(= K, (1-\varepsilon)\cdot D)$ vertex expander. To show that Con is a lossless condenser, it suffices to show that $\mathrm{Con}(X, U_d)$ is ε-close to a k'-source for every *flat* k-source X. Let S be the support of X, which is of size K. By the vertex expansion of the graph, $|N(S)| \geq (1-\varepsilon)\cdot DK$. Since there are only DK edges leaving S, we can make all of these edges lead to distinct vertices by shifting an ε fraction of them. Let $T \subset [M]$ be the set of KD vertices hit after this shifting. Then $\mathrm{Con}(X, U_d)$ is ε-close to the uniform distribution on T, which is a $(k+d)$-source. ■

Thus, the lossless condenser that we assumed in the previous lecture follows immediately from the following expander:

Theorem 3 *For every constant $\alpha > 0$, every $N \in \mathbb{N}$, $K \leq N$, and $\varepsilon > 0$, there is an explicit $(K, (1-\varepsilon)D)$ expander with N left-vertices, M right-vertices, left-degree $D = O((\log N)(\log K)/\epsilon)^{1+1/\alpha}$ and $M \leq D^2 \cdot K^{1+\alpha}$. Moreover, D is a power of 2.*

We will construct this expander after Spring Break, using ideas based on list-decodable error-correcting codes.

Note that the kind of expander given by this theorem can be used for data structure application on Problem Set 3 — storing a $K/2$ subset of S with $M = K^{1+\alpha} \cdot \text{polylog}(N)$ bits in such a way that membership can be probabilistically tested by reading only 1 bit of the data structure. (An efficient solution to this application actually requires more than the fact that the graph is explicit in the usual sense, but also that there are efficient algorithms for finding all left-vertices having a δ fraction neighbors in a set $T \subseteq [M]$ of right vertices, but it turns out that the expanders we will construct turn out to have this property.)

The connection of Lemma 2 is also useful in the reverse direction. Indeed, as we saw last time, there are explicit extractors with entropy loss $\ell = O(\log(1/\varepsilon))$ and seed length $d = O(\log k \cdot \log(n/\varepsilon))$. By applying an almost pairwise-independent hash function, these can be converted into lossless condensers with the same (polylogarithmic) seed length and optimal output length $m = k + d + \log(1/\varepsilon) + O(1)$. These correspond to expanders with expansion $(1-\varepsilon) \cdot D$ whose degree is quasipolynomial in $(\log N)/\varepsilon$, but whose right-hand side is of the optimal size $M = O(KD/\varepsilon)$.

Open Problem 4 *Construct highly unbalanced bipartite expanders with degree $D = \text{poly}(\log N)$, expansion $(1-\varepsilon) \cdot D$ for an arbitrarily small constant $\varepsilon > 0$, and $M = O(KD)$ right-hand vertices.*

Such a construction would give lossless condensers whose output is of extremely high min-entropy ($k' = m - O(1)$), and thus we could get extractors that extract all the min-entropy by then applying the high min-entropy extractor based on spectral expanders.

2 Block-Source Extraction vs. the Zig-Zag Product

Recall the block-source extraction method presented last time. We define $\text{Ext}' : \{0,1\}^{n_1+n_2} \times \{0,1\}^{d_2} \to \{0,1\}^{m_1}$ by $\text{Ext}'((x_1, x_2), y_2) = \text{Ext}_1(x_1, \text{Ext}_2(x_2, y_2))$. (Here we consider the special case that $m_2 = d_1$.

Viewing the extractors as bipartite graphs, the left-vertex set is $[N_1] \times [N_2]$ and the left-degree is D_2. A random step from a vertex $(x_1, x_2) \in [N_1] \times [N_2]$, corresponds to taking a random step from x_2 in G_2 to obtain a right-hand vertex $y_1 \in \{0,1\}^{m_2}$, which we view as an edge label y for G_1. We then move to the y'th neighbor of x_1.

This is just like the first two steps of the zig-zag graph product. Why do we need a third step in the zig-zag product? It is because of the slightly different goals in the two setting. In a (spectral) expander, we consider an arbitrary initial distribution that does not have too much (Renyi) entropy, and need to add entropy to it. In a block-source extractor, our initial distribution is constrained

to be a block source (so both blocks have a certain amount of min-entropy), and our goal is to produce an almost-uniform output (even if we end up with less bits than the initial entropy).

Thus, in the zig-zag setting, we must consider the following extreme cases (that are ruled out for block sources):

- The second block has no entropy given the first. Here, the step using G_2 will add entropy, but not enough to make y_1 close to uniform. Thus, we have no guarantees on the behavior of the G_1-step, and we may lose entropy with it. For this reason, we keep track of the edge used in the G_1-step — that is, we remember b_1 such that x_1 is the b_1'th neighbor of $z_1 = \mathrm{Ext}(x_1, y_1)$. This ensures that the mapping $(x_1, y_1) \mapsto (z_1, b_1)$ is a permutation and does not lose any entropy. We can think of b_1 as 'buffer' that retains any extra entropy in (x_1, y_1) that did not get extracted into z_1. So a natural idea is to just do block source extraction, but output (z_1, b_1) rather than just z_1. However, this runs into trouble with the next case.
- The first block has no entropy but the second block is completely uniform given the first. In this case, the G_2 step cannot add any entropy. But the G_1 step transfers entropy into z_1. So if we add another expander-step from b_1 at the end, we can argue that it will add entropy.

While we analyzed the zig-zag product with respect to spectral expansion (i.e. Renyi entropy), it is also possible to do analyze it in terms of a condenser-like definition (i.e. distributions ε-close to having some min-entropy). However, this construction cannot yield a lossless condenser, because as we have seen, there are cases where only one of the two steps on the 'small' expander(s) 'works' and the entropy of the other one is lost. This can be remedied using a variant of the zig-zag product that keeps buffers also for the steps on the small expander(s). This leads to a construction of constant-degree bipartite expanders with expansion $(1 - \varepsilon) \cdot D$ for the balanced ($M = N$) or slightly unbalanced (e.g. $M = \Omega(N)$) case. However, some important open problems remain.

Open Problem 5 *Construct constant-degree non-bipartite expanders with vertex expansion larger than $D/2$.*

Open Problem 6 *Construct constant-degree expanders (even bipartite ones) with vertex expansion $D - O(1)$.*