Randomness, Pseudorandomness and Derandomization Sept 16th, 2010

Lecture Notes 3: Expander graphs – a ubiquitous pseudorandom structure

Professor: Avi Wigderson (Institute for Advanced Study) *Scribe: Dai Tri Man Lê*

In this lecture, we will focus on expander graphs (also called expanders), which are pseudorandom objects in a more restricted sense than what we saw in the last two lectures. The reader is also referred to the monograph [1] and the tutorial slides [2] for more detailed surveys of today's topics.

Expander graphs are universally useful in computer science and have many applications in derandomization, circuit complexity, error correcting codes, communication and sorting networks, approximate counting, computational information, data structures, etc. Expander graphs also have many interesting applications in various areas of pure mathematics: topology, group theory, measure theory, number theory and especially graph theory.

1 Expander graphs: definition and basic properties

Expander graphs are graphs with an additional special "expansion" property. This property can be equivalently described combinatorically, geometrically, probabilistically, or algebraically.

1. **Combinatoric formulation:** Expander graphs are sparse d-regular graph (for some fixed d), but highly connected; any two disjoint sets of vertices cannot be disconnected from each other without removing a lot of edges (i.e., no small cuts). In other words, a d-regular graph G with n vertices is expander if for any subset S of vertices satisfying $|S| < n/2$, number of edges connecting S and the complement of S is at least $\alpha|S|d$ for a constant α.
2. **Geometric formulation:** Expander graphs have high isoperimetry, i.e., any proper subset S of vertices has a large boundary ∂S, where ∂S is defined as the set of vertices which are not in S but are adjacent to a vertex in S.
3. **Probabilistic formulation:** A random walk on an expander graph converges very rapidly (in $O(\log n)$ steps) to the uniform distribution.
4. **Algebraic formulation:** If we look at the the adjacency matrix of an expander graph, then there is a large gap between second largest eigenvalue and the largest eigenvalue, which in this case equals the degree of the graph.

Since all of these formulations of the expansion property are equivalent, we will use the algebraic formulation in this lecture. Let $G = (V, E)$ be a d-regular graph with n vertices. We let A_G denote the normalized adjacency matrix of G, i.e.,

$$A_G(u, v) = \begin{cases} 0 & \text{if } (u, v) \in E \\ 1/d & \text{if } (u, v) \notin E \end{cases}$$

It easy to check that if we sort the eigenvalues λ_i of A_G in non-increasing order, then $1 \geq \lambda_i \geq -1$ and the largest eigenvalue $\lambda_1 = 1$. We will let $\lambda(G)$ denote the second largest eigenvalue of A_G. We call $1 - \lambda(G)$ the *spectral gap* of G.

Definition 1. *A graph G is an $[n, d]$-graph if it is a d-regular graph with n vertices. A graph G is an $[n, d, \delta]$-graph if it is an $[n, d]$-graph satisfying $\lambda \leq \delta$. An infinite family $\{G_i\}$ of $[n_i, d, \delta]$-graphs forms an expander family if $0 < \delta < 1$ for all i.*

It can be showed that most d-regular graphs (even when $d = 3$) are expanders, but in almost all applications, the main challenge is to construct (small degree) expanders efficiently and explicitly.

As mentioned above, expander graphs are pseudorandom objects. From the previous two lectures, we know that an object is pseudorandom if it looks random to all "efficient" algorithms. Expanders are pseudorandom in a much more limited sense since it can only fool the *cut* function for graphs. Let $G = (V, E)$ be an $[n, d, \delta]$-expander. Given two disjoint sets of vertices $S, T \subseteq V$, let $E(S, T)$ denote the set of edges from S to T, and thus $E(S, T)$ is the cut-set of two partitions S and T. We can show the following theorem

Theorem 1. *For any two disjoint sets of vertices $S, T \subseteq V$,*

$$|E(S,T)| = \frac{d|S||T|}{n} \pm \delta dn.$$

Note that $\frac{d|S||T|}{n}$ is just the expectation of the size of the cut-set $|E(S, T)|$ in a random graph, and δdn is some small error depending on the parameter δ of G. From Theorem 1, we have the following corollaries.

Corollary 1. *Every vertex set of size greater than δn contains an edge. And thus the chromatic number of an $[n, d, \delta]$-expander G is at least $1/\delta$. Thus, expander graphs can be used to construct graphs with large girth or large chromatic number.*

Corollary 2. *Removing any fraction $\gamma < \delta$ of the edges of an $[n, d, \delta]$-expander G leaves a connected component consisting of $1 - O(\gamma)$-fraction of the vertices, and thus G is still highly connected.*

2 Some Applications of expander graphs

2.1 Network reliability

Expanders have many applications in designing fault-tolerant networks and distributed systems. These applications are based mainly on the following corollary that can be easily shown from the pseudorandomness property in Theorem 1.

Corollary 3. *Given an $[n, d, \delta]$-expander G with $\delta < 1/4$, every set S of vertices of G of size at most $\delta n/2$ contains at most $s/2$ vertices with a majority of neighbors in S.*

Consider the following infection processes on a network represented by a $[n, d, \delta]$-expander G with $\delta < 1/4$.

1. **Infection process 1:** Assume that adversary infect an infection set I_0 of vertices with $|I_0| \leq \delta n/4$. Then the infection process can be represented by a sequence of sets:

   $$I_0 = S_0, S_1, \ldots, S_t, \ldots,$$

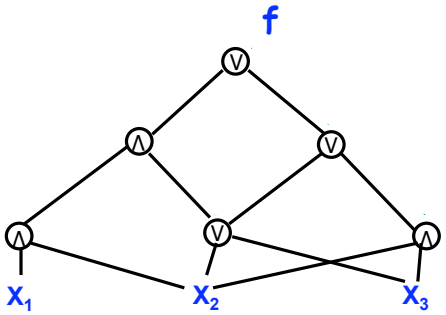
   where a vertex $v \in S_{t+1}$ iff a majority of its neighbors are in S_t. Then it follows from Corollary 3 that the size of S_t reduce exponentially and so $S_t = \varnothing$ for $t > \log n$. In words, the infection dies out very fast.

2. **Infection process 2:** Assume that adversary infect the infection sets $I_0, I_1, I_2, \ldots$, with $|I_t| \leq \delta n/4$, and the infection process is defined as

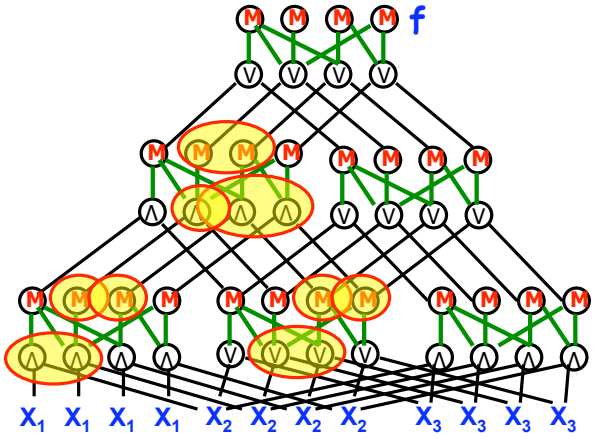
$$I_0 = R_0, R_1, \ldots, R_t, \ldots,$$

where a vertex $v \in R_{t+1}$ iff a majority of its neighbors are in R_t or $v \in I_{t+1}$. Note that we have a new supply of infection I_t in every round. However, it can be shown from Corollary 3 that $R_t \leq \delta n/2$ for all t, and so the infection never spreads.

To demonstrate how these infection-process properties of expanders are used, we will consider an example of how to construct reliable circuits from unreliable components, which is a problem studied von Neumann in the 60s. Assume we are given a circuit C computing a boolean function f of size s such that every gate fails with probability $p < 1/10$. Goal: we want to construct a circuit C' with size s' not much bigger than s such that $C'(x) = f(x)$ with high probablity.



It was shown by von Neumann, Dobrushin-Ortyukov and Pippenger that a much more reliable construction of C' can be achieved by replicating the circuit C and using majority "expanders" to connect different copies of each gate as shown in the following figure.



The intuition is that this construction of C' will take care of the errors as in the 2nd infection process discussed above. Since each majority is connected many neighbors below it, even when some errors are produced in some of the neighbors, we expect the errors will die out and will not propagate so much to the output gates of C'.

2.2 Deterministic error reduction

As mentioned in the first lecture, a probabilistic algorithm $A(x, r)$ takes some input x and also some random input $r \in \{0,1\}^n$. Assume that for every input x, this algorithm return the correct answer for $2/3$ of the random inputs $r \in \{0,1\}^n$ and return the wrong answer for the other $1/3$ of the random inputs. Then we can reduce the failure probability by choosing k independent random bit string $r_1, \ldots, r_k \in \{0,1\}^n$, and runs the algorithm A once for each such random string, and then takes the majority vote of the k outcomes. It follows from a simple application of Chernoff's bound that the failure probability of this new repetition algorithm is exponentially small. The only disvantage is that this algorithm requires km independent random bits instead of m random bits as in the original algorithm. It turns out that we can obtain an algorithm with similar performance but use many fewer random bits. We first need an explicit construction of a $[2^n, d, 1/8]$-expander G, whose vertices represent 2^n possible random strings in $\{0,1\}^n$, and we can simply select $r_1, \ldots, r_k$, which are now vertices of G, in turn by taking a random walk on G starting from a random vertex, and then perform a majority vote of the outcomes. The rapid mixing properties of random walk on expanders can be used to show the same type of exponentially small failure probability as before, but now only $n + O(k)$ random bits are needed.

2.3 Metric embeddings

We write (X, d) to denote a metric space on the set of points X equipped with metric d. A metric space (X, d) embeds with distortion Δ into the ℓ_2 space (Hilbert space) if there exist an injective function $f : X \to \ell_2$ such that for all points $x, y \in X$,

$$d(x, y) \leq \|f(x) - f(y)\|_2 \leq \Delta\, d(x, y).$$

Theorem 2 (Bourgain). *Every n-point metric space has an embedding with distortion $O(\log n)$ into the ℓ_2 space.*

Linial-London-Rabinovich actually showed that this is tight when we let (X, d) be the distance metric of an appropriate expander graph. Their proof uses basic algebraic properties of expanders and Poincaré's inequality.

Now we look at a more liberal notion of metric embedding. A metric space has a coarse embedding into the ℓ_2 space if an injective function $f : X \to \ell_2$ and increasing, unbounded function $\phi, \sigma : \mathbb{R} \to \mathbb{R}$ such that for all $x, y \in X$,

$$\phi(d(x, y)) \leq \|f(x) - f(y)\|_2 \leq \sigma(d(x, y)).$$

One might ask if there exists a class of metric spaces which does not have a coarse embedded into the ℓ_2 space. Gromov showed that, in fact, there exists a finitely generated and finitely presented group, who Cayley graph metric has no coarse embedding into the ℓ_2 space. His proof uses an infinite sequence of Cayley expander graphs. This result is very relevant to the Novikov and Baum-Connes conjectures since it might give counter-examples to refute these conjectures.

3 Explicit constructions of expander graphs

In many applications mentioned in the previous section, we need to construct suitable expander graphs explicitly. There are both algebraic and combinatorial explicit constructions of expanders.

3.1 Algebraic constructions

The first explicit expander families arose from group theory (as given by Margulis, Gabber-Galil, Alon-Milman, Lubotsky-Phillips-Sarnak, and others); a typical example of such a family are the Cayley graphs, where vertices are elements of the special linear group $SL_2(p)$ (i.e., group of 2×2 matrices, whose entries are taken over a large field of prime order p and whose determinants equal to 1) and the edges are given by the set of generators

$$S := \left\{ \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \right\}.$$

Thus, two matrices in $SL_2(p)$ are connected by a graph if their quotient is either in S, or is an inverse of an element in S. The expansion of these algebraic graph families is deeply connected to some non-trivial results in group theory and analytic number theory, in particular, the famous Selberg's $3/16$ theorem on the least eigenvalue of the Laplacian on modular curves.

The construction of the Cayley expanders mentioned here is strongly explicit: if we need n bits to describe a matrix M in $SL_2(p)$ (assuming that the expander graph has $\mathsf{exp}(n)$ vertices), then we can compute the 4 neighbors of M in $\mathsf{poly}(n)$ time! In many cases, we can even compute the neighborhoods in logspace.

These algebraic constructions can be made extremely tight in the sense that the parameter δ can be made optimally small. More precisely, there is the Alon-Boppana bound, which asserts that an $[n, d, \delta]$-graph can only exist if $d\delta \geq 2\sqrt{d-1}$. Graphs which attain this bound are known as Ramanujan graphs, and were first constructed by Lubotsky-Phillips-Sarnak and Margulis by algebraic methods.

3.2 Zigzag graph product and combinatorial constructions

Starting with the work of Reingold, Vadhan, and Wigderson in 2000, explicit combinatorial constructions of expanders have been given.

The zigzag product. A fundamental building block in combinatorial constructions is the *zigzag product* $G \textcircled{z} H$ of an $[n, m, \alpha]$-graph G and an $[m, d, \beta]$-graph H, where H is small compared to G. It is important to note that the number of vertices of in the small graph H equals the degree of the large graph G, and thus we can "blow up" each vertex of G to into a copy of H and decouple the edges in G into disjoint edges. Let V and W be the vertex sets of G and H respectively, then $G \textcircled{z} H$ is defined to be the $[nm, d+1]$-graph, where

- the vertex set $U = \{(v, k) \mid v \in V \wedge k \in W\}$, and
- $((u, k), (v, \ell)) \in U \times U$ is an edge of $G \textcircled{z} H$ if there exists a "three step sequence" consisting of an H-step from (u, k) to (u, i) (corresponding to the edge $\{k, i\} \in H$), followed by a G-step from (u, i) to (v, j) (corresponding to the edge $\{u, v\} \in G$), followed by H-step from (v, j) to (v, ℓ) (corresponding to the edge $\{j, \ell\} \in H$).

It can be shown from the zigzag product construction that:

Theorem 3 (Reingold-Vadhan-Wigderson). *The zigzag product $G \textcircled{z} H$ is an $[nm, d^2, \alpha + \beta]$-graphs, and thus $G \textcircled{z} H$ is an expander if G and H are.*

Note that the degree $G \textcircled{z} H$ is d^2, which is small and independent of the degree of the large graph G. Thus, G and H are expanders and the small graph H has small degree, then zigzag product produces another bigger expander with also small degree.

Iterative construction of expanders The iterative construction of expanders combines both the zigzag product and the graph squaring. The construction starts with a constant size H, which is a $[d^4, d, 1/4]$-graph. Then define

$$G_1 = H^2$$
$$G_{k+1} = G_k^2 ⓩ H$$

From this construction and Theorem 3, we have the following theorem.

Theorem 4 (Reingold-Vadhan-Wigderson). *The sequence $\{G_k\}$ is an finite sequence of expanders, and G_k is an $[d^{4k}, d^2, 1/2]$-graph.*

The zigzag product has many interesting consequences. Recall that the algebraic construction gave us very tight spectral bounds, but it turns out that they do not give us very tight isoperimetric bounds. Using the idea of zigzag product, Reingold-Vadhan-Wigderson and Capalbo-Reingold-Vadhan-Wigderson were able to achieve much better isoperimetric bounds. The zigzag product can also be seen as a combinatorial generalization of semi-direct product in groups as shown in the work of Alon-Lubotzky-Wigderson. Using this result, new constructions of expanding Cayley graphs for *non-simple* groups were introduced by Meshulam-Wigderson and Rozenman-Shalev-Wigderson.

One particularly striking recent application of the zig-zag product construction (by Reingold in 2005) is to create a deterministic logarithmic space algorithm for reachability problem for undirected graphs (also known as the maze exploration problem as mentioned in the previous lecture). This resolved a 25-year open problem in complexity theory, and showed that $\mathsf{L} = \mathsf{SL}$.

Another place where zigzag product has been used is in the construction of lossless expanders. Given an $[n, d]$-graph, the neighborhood of a k-vertex set can have size at most dk. A *lossless expander* is a family of $[n, d]$-graphs with the property that for any $\varepsilon > 0$, there exists a c such that every vertex set S with $|S| \leq n/c$ has a neighborhood of size at least $(1 - \varepsilon)d|S|$; thus small sets expand nearly as much as possible. Ramanujan graphs do not always have this property since no matter how large one makes c, one can find such graphs in which the neighbourhood of a vertex set S is at most $d|S|/2$. But by using the zigzag construction (this time to a type of graph known as a randomness conductor) one can create lossless expanders.

4 Lossless expanders and error-correcting codes

We will end this lecture by showing one application of lossless expanders to error-correcting codes. A code is an injective function $C : \{0,1\}^k \to \{0,1\}^n$. The list of codewords is the image of C (which is often denoted by C by abuse of notation). There are many important statistics of codes, but two particularly key ones are the rate $\mathsf{Rate}(C) = k/n$, and the Hamming distance $\mathsf{Dist}(C)$, which is the minimal separation between two codewords. A code is good if $\mathsf{Rate}(C) = \Omega(1)$, and if the Hamming distance $\mathsf{Dist}(C) = \Omega(n)$.

It was already shown by Shannon in 1948 using probabilistic method that good codes exist, indeed a randomly selected code is likely to be rather good! On the other hand, it is not possible to perform error correction on a random code quickly. Thus, the main challenge is to find good, explicit and efficient codes. There many explicit algebraic constructions of codes: Hamming, BCH, Reed-Solomon, Reed-Muller, Goppa, etc., and many explicit combinatorial constructions: Gallager, Tanner, Luby-Mitzenmacher-Shokrollahi-Spielman, Sipser-Spielman, etc.

One remarkable result achieved by combinatorial construction, and not algebraic, is the code due to Spielman, which is good, explicit and has $O(n)$ time encoded and decoding algorithms. We will next give the basic idea of this combinatorial code construction.

Recall that a code is *linear* if C is a linear function (identifying $\{0,1\} \equiv \mathbb{F}_2$ with the field of two elements). Thus, the image of C is a linear subspace of $\mathbb{F}_2^n$, and thus can be defined as the simultaneous null space of $n-k$ linear functionals on this space, which can also be viewed as checksums or parity bits that need to vanish in order for a given word to lie in the code. Over $\mathbb{F}_2$, a linear functional on the Hamming cube $\{0,1\}^n \equiv \mathbb{F}_2^n$ is nothing more than the sum of some collection of the coordinate functions on that cube. Thus one can describe the code C by a bipartite graph connecting the n coordinates with the $n-k$ parity bits, with each parity bit being formed as the sum of the coordinates it is connected to.

With linear codes, encoding is a relatively quick process; the challenge is in decoding in the presence of errors. Remarkably, if one selects the graph defining the code to be a bipartite lossless expander (so that each of the n coordinates is connected to d parity bits for some bounded d, and any m coordinates, with m less than a small multiple of n, is connected to close to dm parity bits), then not only is the code good, one can decode in linear time by the following belief propagation algorithm:

1. Start with the corrupted word, and compute all the parity bits. If they all vanish, then we accept the codeword and we stop.
2. Otherwise, we go through each coordinate and see how many of the d parity bits connected to that coordinate are nonzero (i.e. they take the value 1). If a majority of these bits return 1, then we flip the bit of the coordinate; otherwise, we keep the bit unchanged. We repeat this step until all parity bits vanish.

Intuitively, this algorithm works since in a lossless expander, the parity bits associated to each of the corrupted coordinates are mostly disjoint. Thus, every corrupted coordinates should see that most of the parity bits connected to it are taking the value of 1, giving a strong signal that those bits should change. Conversely, the uncorrupted bits should mostly see only a small minority of parity bits connected to it reporting the value 1. Indeed, one can show from the properties of lossless expanders that each iteration of Step 2 cuts down the number m of corrupted bits by a constant factor. Thus, the run-time of this algorithm is linear in the length n of the code.

(End of the third talk.)

Acknowledgment Some parts of the lecture notes follow an exposition by Terry Tao on a similar talk given by Avi Wigderson in UCLA.

References

1. S. Hoory, N. Linial , A. Wigderson. Expander Graphs and their Applications. Bull. Amer. Math Soc., 43, pp 439–561, 2006. Also available at: http://www.math.ias.edu/∼avi/BOOKS/expanderbookr1.pdf
2. A. Wigderson. *Expander graphs - applications and combinatorial constructions.* A 3-hour tutorial, Pseudorandomness in Mathematical Structures Workshop, IAS, Princeton, NJ - June 14-18, 2010. Slides available at: http://www.math.ias.edu/ avi/TALKS/expander_tutorial_June2010.ppt