Expander Graphs

Prof. Jara Uitto *Antonio Cruciani*

1 Magical graphs and saving randomness

The examples so far show that expanders are sparse graphs with strong connectivity properties. We now see a first algorithmic application: expansion can be used to reduce the number of random bits needed by a randomized algorithm.

The main idea is that if a graph has good expansion, then a small random neighborhood behaves like several random samples. This lets us replace many independent random choices by one random choice followed by looking at its neighbors.

Magical graphs. Let $G = (L, R, E)$ be a bipartite graph with

$$|L| = n, \qquad |R| = m,$$

where every vertex in L has degree d. For a set $S \subseteq L$, let $\Gamma(S) \subseteq R$ be the set of neighbors of S.

We say that G is an $(n, m; d)$-magical graph if the following two properties hold:

1. For every $S \subseteq L$ with
$$|S| \leq \frac{n}{10d},$$
we have
$$|\Gamma(S)| \geq \frac{5d}{8}|S|.$$
2. For every $S \subseteq L$ with
$$\frac{n}{10d} < |S| \leq \frac{n}{2},$$
we have
$$|\Gamma(S)| \geq |S|.$$

The definition says that small sets expand by a factor proportional to d, while medium-size sets do not shrink. This is a vertex-expansion condition for a bipartite graph.

Theorem 1. *For every sufficiently large constant d, magical graphs exist. More precisely, for every $d \geq 32$, every sufficiently large n, and every $m \geq 3n/4$, there exists an $(n, m; d)$-magical graph.*

Proof idea. Choose a random bipartite graph with left side L, right side R, and let every vertex of L choose d random neighbors in R.

We want to show that, with positive probability, no set $S \subseteq L$ has too small a neighborhood. For fixed sets $S \subseteq L$ and $T \subseteq R$, the probability that all edges from S land in T is

$$\left(\frac{|T|}{m}\right)^{d|S|}.$$

We then take a union bound over all bad choices of S and T. For sufficiently large constant d, the total failure probability is less than 1. Hence there exists a graph satisfying the desired expansion properties. $\square$

Application: saving random bits. Suppose that L is a language with a one-sided-error randomized algorithm A. On input x, the algorithm uses k random bits $r \in \{0,1\}^k$. If $x \in L$, then

$$A(x,r) = 1$$

for every r. If $x \notin L$, then

$$\Pr_{r \in \{0,1\}^k}(A(x,r) = 1) < \frac{1}{16}.$$

The standard way to reduce the error is to run the algorithm many times with independent random strings. This reduces the error, but it also uses many more random bits.

We now use a magical graph to save randomness. Let

$$n = 2^k,$$

and identify both sides of an $(n, n; d)$-magical graph with the set

$$\{0,1\}^k.$$

To decide whether $x \in L$, do the following:

1. Choose one random string $r \in \{0,1\}^k$. Think of r as a vertex on the left side of the magical graph.
2. Let
$$r_1, \ldots, r_d$$
be the d neighbors of r on the right side.
3. Output 1 if and only if
$$A(x, r_i) = 1$$
for every $i = 1, \ldots, d$.

Proposition 2. *The new algorithm has one-sided error and fails with probability at most*

$$\frac{1}{10d}.$$

It uses only k random bits.

Proof. If $x \in L$, then $A(x, r_i) = 1$ for every random string r_i. Therefore the new algorithm always accepts.

Now suppose that $x \notin L$. Let

$$B = \{r \in \{0,1\}^k : A(x,r) = 1\}$$

be the set of bad random strings, namely the strings on which the original algorithm incorrectly accepts. By assumption,

$$|B| < \frac{n}{16}.$$

The new algorithm fails exactly when all neighbors of the chosen left vertex r lie in B. Let $S \subseteq L$ be the set of left vertices with this property:

$$S = \{r \in L : \Gamma(r) \subseteq B\}.$$

Then

$$\Gamma(S) \subseteq B.$$

We claim that

$$|S| \leq \frac{n}{10d}.$$

Indeed, suppose for contradiction that

$$|S| > \frac{n}{10d}.$$

Then we can choose a subset $S' \subseteq S$ of size exactly $n/(10d)$ up to rounding. By the first property of magical graphs,

$$|\Gamma(S')| \geq \frac{5d}{8}|S'| = \frac{5d}{8} \cdot \frac{n}{10d} = \frac{n}{16}.$$

But since $S' \subseteq S$, we have

$$\Gamma(S') \subseteq \Gamma(S) \subseteq B.$$

This contradicts

$$|B| < \frac{n}{16}.$$

Therefore,

$$|S| \leq \frac{n}{10d}.$$

The algorithm fails exactly when the initially chosen random vertex $r \in L$ lies in S. Since r is uniform in L,

$$\Pr(\text{failure}) = \frac{|S|}{n} \leq \frac{1}{10d}.$$

□

The usual amplification method uses d independent random strings and therefore dk random bits. The magical-graph method uses only one random string, namely k random bits, and then replaces independent samples by the neighbors of that string in an expanding graph.

This is the first glimpse of a broader principle: random walks and neighborhoods in expanders can imitate independent random samples while using far fewer random bits.

Explicitness matters. There is an important algorithmic point hidden in the construction. The vertices of the magical graph correspond to random strings

$$r \in \{0,1\}^k.$$

Thus the graph has

$$n = 2^k$$

vertices, which is exponential in the number of random bits used by the original algorithm.

Therefore, for the amplification procedure to be efficient, we cannot store the whole graph in memory. Instead, we need a very explicit description of the graph: given

$$r \in \{0,1\}^k \qquad \text{and} \qquad i \in \{1,\dots,d\},$$

there should be a deterministic procedure

$$\mathrm{Nbr}(r,i)$$

that returns the i-th neighbor of r in time polynomial in $k = \log n$.

The randomness-efficient amplification algorithm is then:

Input:	x
Random choice:	choose one string $r \in \{0,1\}^k$ uniformly at random
For $i = 1,\dots,d$:	$r_i := \mathrm{Nbr}(r,i)$
	run the original algorithm $A(x,r_i)$
Output:	1 if $A(x,r_i) = 1$ for every $i = 1,\dots,d$; 0 otherwise.

The usual amplification method would choose

$$r_1,\dots,r_d \in \{0,1\}^k$$

independently, using dk random bits. Here we choose only one random string r, using k random bits, and generate $r_1,\dots,r_d$ deterministically as the neighbors of r.

The expansion property guarantees that the set of left vertices whose entire neighborhood is bad is small. Therefore, although the strings $r_1,\dots,r_d$ are not independent, they are sufficiently well spread out to reduce the error.

Expansion lets us trade independence for structure. Instead of using many independent random strings, we use one random string and the expanding neighborhood around it. This saves randomness while still reducing the error probability.

2 Expanders are resilient to adversarial faults

We now show another important reason why expanders are useful: they are resilient to faults. Even if an adversary removes some vertices, a large part of the graph still contains a well-expanding subgraph.

In this section we use vertex expansion. For a graph $G = (V, E)$ and a set $S \subseteq V$, define

$$\Gamma_G(S) = \{v \in V \setminus S : \text{there exists } u \in S \text{ with } \{u, v\} \in E\}.$$

The vertex expansion of G is

$$\alpha(G) = \min_{\substack{S \subseteq V \\ 0<|S|\leq|V|/2}} \frac{|\Gamma_G(S)|}{|S|}.$$

Suppose G has vertex expansion α, and an adversary removes some faulty vertices. Let G_f be the remaining graph. The graph G_f may no longer be an expander: the adversary may create small pieces with poor expansion. The idea is to prune away these bad pieces until the remaining graph has good expansion again.

The pruning procedure. Let $0 < \eta < 1$. Define the following procedure.

Prune(η) :
  $G_0 \leftarrow G_f, \quad i \leftarrow 0.$
  while there exists $S_i \subseteq V(G_i)$ **such that** $|S_i| \leq |V(G_i)|/2$ **and**
    $|\Gamma_{G_i}(S_i)| \leq \eta\alpha|S_i|$ **do**
    $G_{i+1} \leftarrow G_i \setminus S_i.$
    $i \leftarrow i + 1.$
  end while
  $H \leftarrow G_i.$

The procedure repeatedly removes sets whose expansion has become too small. When the procedure stops, no such bad set remains, so the remaining graph H has vertex expansion at least $\eta\alpha$.

Intuition. Each time the pruning procedure removes a poorly expanding region, that region must have lost many of its original neighbors to faults. Since the adversary has only f faulty vertices available, the total amount of material that can be pruned away is limited. Therefore a large expanding core remains.

Theorem 3 (Resilience under adversarial faults)**.** *Let $G = (V, E)$ be a graph with n vertices, maximum degree $\Delta \geq 2$, and vertex expansion $\alpha > 0$. Let $F \subseteq V$ be a set of faulty vertices, with*

$$|F| = f \leq \frac{\alpha n}{4\Delta k^2},$$

where $k > 1$ is a constant. Let $G_f = G[V \setminus F]$ be the graph obtained after removing the faulty vertices.

Then

$$\textsc{Prune}\left(1 - \frac{1}{k}\right)$$

returns a subgraph $H \subseteq G_f$ such that

$$\alpha(H) \geq \left(1 - \frac{1}{k}\right)\alpha$$

and

$$|V(H)| \geq n - f - \frac{kf}{\alpha}.$$

Proof. Let

$$\eta = 1 - \frac{1}{k}.$$

By the stopping condition of the pruning procedure, the output graph H has vertex expansion at least $\eta\alpha$. Indeed, if some non-empty set $S \subseteq V(H)$ with $|S| \leq |V(H)|/2$ satisfied

$$|\Gamma_H(S)| \leq \eta\alpha|S|,$$

then the algorithm would not have stopped.

It remains to prove that not too many non-faulty vertices are removed. Let

$$R := V(G_f) \setminus V(H)$$

be the union of all sets removed by the pruning procedure. We prove that

$$|R| \leq \frac{kf}{\alpha}.$$

Since G_f has $n - f$ vertices, this implies

$$|V(H)| = |V(G_f)| - |R| \geq n - f - \frac{kf}{\alpha}.$$

Let $S_0, S_1, \ldots$ be the sets removed by the pruning procedure, and for $j \geq 0$ define

$$R_j = \bigcup_{i=0}^{j} S_i.$$

We first record the key observation:

$$|\Gamma_{G_f}(R_j)| \leq \sum_{i=0}^{j} |\Gamma_{G_i}(S_i)| \leq \eta\alpha|R_j|.$$

Indeed, every vertex $v \in \Gamma_{G_f}(R_j)$ is adjacent in G_f to some removed set S_i. Since $v \notin R_j$, the vertex v has not been removed in any of the rounds $0, \ldots, j$. Hence, if v is adjacent to some S_i, then v was still present when S_i was removed, and so

$$v \in \Gamma_{G_i}(S_i).$$

Therefore,

$$\Gamma_{G_f}(R_j) \subseteq \bigcup_{i=0}^{j} \Gamma_{G_i}(S_i).$$

It follows that

$$|\Gamma_{G_f}(R_j)| \leq \sum_{i=0}^{j} |\Gamma_{G_i}(S_i)|.$$

By the pruning rule,

$$|\Gamma_{G_i}(S_i)| \le \eta\alpha|S_i|.$$

Since the removed sets are disjoint,

$$\sum_{i=0}^{j} |S_i| = |R_j|.$$

Thus

$$|\Gamma_{G_f}(R_j)| \le \eta\alpha|R_j|.$$

Now suppose, for contradiction, that

$$|R| > \frac{kf}{\alpha}.$$

Since each removed set has size at most half of the graph at the moment it is removed, there is an index j for which one of the following two cases occurs.

Case 1. There exists j such that

$$\frac{kf}{\alpha} < |R_j| \le \frac{n}{2}.$$

Since G has vertex expansion α,

$$|\Gamma_G(R_j)| \ge \alpha|R_j|.$$

On the other hand,

$$|\Gamma_{G_f}(R_j)| \le \eta\alpha|R_j| = \left(1 - \frac{1}{k}\right)\alpha|R_j|.$$

Thus at least

$$\alpha|R_j| - \left(1 - \frac{1}{k}\right)\alpha|R_j| = \frac{\alpha}{k}|R_j|$$

vertices from the original boundary of R_j are missing in G_f. All these vertices must be faulty. Hence

$$f \ge \frac{\alpha}{k}|R_j| > \frac{\alpha}{k} \cdot \frac{kf}{\alpha} = f,$$

a contradiction.

Case 2. There exists j such that

$$|R_{j-1}| \le \frac{kf}{\alpha}$$

and

$$\frac{n}{2} - \frac{kf}{\alpha} < |S_j| \le \frac{n}{2}.$$

Let

$$R' = R_{j-1}.$$

From the key observation,

$$|\Gamma_{G_f}(R')| \le \eta\alpha|R'| \le \alpha|R'| \le kf.$$

Thus at most kf vertices of S_j can have a neighbor in R'. Since the maximum degree is Δ, the number of vertices in R' adjacent to S_j is at most

$$\Delta kf.$$

When S_j is removed, we have

$$|\Gamma_{G_j}(S_j)| \leq \left(1 - \frac{1}{k}\right) \alpha |S_j|.$$

The only additional neighbors of S_j in G_f are vertices removed earlier, namely vertices in R'. Therefore,

$$|\Gamma_{G_f}(S_j)| \leq \left(1 - \frac{1}{k}\right) \alpha |S_j| + \Delta k f.$$

But since $|S_j| \leq n/2$, expansion in G gives

$$|\Gamma_G(S_j)| \geq \alpha |S_j|.$$

Hence the number of faulty vertices in the original boundary of S_j is at least

$$\alpha |S_j| - \left(\left(1 - \frac{1}{k}\right) \alpha |S_j| + \Delta k f\right) = \frac{\alpha}{k} |S_j| - \Delta k f.$$

Therefore,

$$f \geq \frac{\alpha}{k} |S_j| - \Delta k f.$$

Using

$$f \leq \frac{\alpha n}{4 \Delta k^2},$$

we get

$$\frac{kf}{\alpha} \leq \frac{n}{4 \Delta k} \leq \frac{n}{8},$$

where we used $\Delta \geq 2$ and $k > 1$. Hence

$$|S_j| > \frac{n}{2} - \frac{kf}{\alpha} \geq \frac{n}{2} - \frac{n}{8} = \frac{3n}{8}.$$

Also,

$$\Delta k f \leq \frac{\alpha n}{4k}.$$

Therefore,

$$f \geq \frac{\alpha}{k} |S_j| - \Delta k f > \frac{\alpha}{k} \cdot \frac{3n}{8} - \frac{\alpha n}{4k} = \frac{\alpha n}{8k}.$$

But

$$f \leq \frac{\alpha n}{4 \Delta k^2} < \frac{\alpha n}{8k},$$

again because $\Delta \geq 2$ and $k > 1$. This is a contradiction.

Both cases are impossible. Therefore

$$|R| \leq \frac{kf}{\alpha}.$$

Consequently,

$$|V(H)| \geq n - f - \frac{kf}{\alpha}.$$

Together with the stopping condition, this proves

$$\alpha(H) \geq \left(1 - \frac{1}{k}\right) \alpha.$$

□

Corollary 4. *Taking $k = 2$, if*

$$f \leq \frac{\alpha n}{16\Delta},$$

then after removing f adversarially chosen vertices, the remaining faulty graph contains a subgraph H with

$$\alpha(H) \geq \frac{\alpha}{2}$$

and

$$|V(H)| \geq n - f - \frac{2f}{\alpha}.$$

References

[1] Trevisan Luca, *Lecture Notes on Graph Partitioning, Expanders and Spectral Methods*,

[2] Hoory, Shlomo, Linial Nathan, Wigderson Avi, *Expander graphs and their applications*,

[3] Bagchi Amitabha, Bhargava Ankur, Chaudhary Amitabh, Eppstein David, Scheideler Christian, *The Effect of Faults on Network Expansion*,

[4] Mitzenmacher, Michael and Upfal, Eli, *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis, Cambridge university press, 2017*,