Dependent random choice, statistical physics, and local rank

Talk by Daniel Zhu

Notes by Sanjana Das

October 31, 2024

§1 Degrees in bipartite graphs

To begin with, we'll talk about an abstract combinatorial scenario. Suppose that we have a bipartite graph G with vertex sets A (on the left) and B (on the right), with $|A| = |B| = n$. We'll say that the edge density of G is δ, so that $|E(G)| = \delta n^2$. We think of δ as small, but n as much larger than δ is small — specifically, we have $\delta = n^{-o(1)}$.

It's a pretty natural scenario that we have a bipartite graph and want to find structure inside it. To do so, we generally care about the degrees of the graph. The *typical* degree of a vertex is δn, so there are two simple statements that we might *want* to be true.

(1) For most $v \in A$, we have $\deg(v) = O(\delta n)$.

(2) For most $v \in A$, we have $\deg(v) = \Omega(\delta n)$.

(Throughout this talk, you can think of 'most' as 90%.)

The first statement is true — the average of all the degrees is δn, so by Markov's inequality you can't have degrees much larger than this too often. But the second statement is false — it's possible that you have a couple of vertices with very high degree, and the remaining degrees are much smaller.

More explicitly, to prove the first statement, we have $\mathbb{E}_{v \in A}[d(v)] = \delta n$, and we can then apply Markov. As foreshadowing, to prove something like the second statement, we'd instead want to consider

$$\mathbb{E}_{v \in A}\left[\frac{1}{d(v)}\right].$$

And you can't control this just knowing δ — you can show that it's *small*, but in order to prove a lower bound of $\Omega(\delta n)$ you'd need to show that it's *large*, and that's not true in general.

So you can't prove (2) in general. But here's an interesting fact that *is* true.

Fact 1.1 — If G has no isolated vertices, then we have

$$\mathbb{E}_{vw \in E(G)} \left[\frac{1}{d(v)} \right] = \frac{1}{\delta n}.$$

(Throughout the talk, we'll assume the relevant graphs G have no isolated vertices.)

Proof. If we imagine summing $\frac{1}{d(v)}$ over all edges vw, every vertex $v \in A$ has $d(v)$ edges incident to it, and each contributes $\frac{1}{d(v)}$, so in total we get

$$\sum_{vw \in E(G)} \frac{1}{d(v)} = \sum_{v \in A} d(v) \cdot \frac{1}{d(v)} = n.$$

And there are δn^2 edges, so

$$\mathbb{E}_{vw \in E(G)} \left[\frac{1}{d(v)} \right] = \frac{n}{\delta n^2} = \frac{1}{\delta n}.$$ □

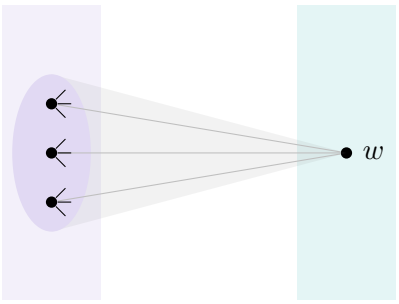
Why is this interesting? We can imagine sampling w first (according to its marginal distribution). Then when we condition on w, the distribution of v is uniform over the neighbors of w. So we can rewrite this as

$$\mathbb{E}_w \left[\mathbb{E}_{v \in N(w)} \left[\frac{1}{d(v)} \right] \right] = \frac{1}{\delta n}.$$

This means there is some w such that

$$\mathbb{E}_{v \in N(w)} \left[\frac{1}{d(v)} \right] \leq \frac{1}{\delta n}.$$

And this is the quantity we wanted to control in order to get a lower bound on degrees — so this means that most $v \in N(w)$ have $d(v) = \Omega(\delta n)$.



We can actually do a bit better. Suppose that instead of just considering $\frac{1}{d(v)}$, we consider $\frac{1}{d(v)} + \frac{1}{d(w)}$. By symmetry and the linearity of expectation, we have

$$\mathbb{E}_w \left[\mathbb{E}_{v \in N(w)} \left[\frac{1}{d(v)} + \frac{1}{d(w)} \right] \right] = \frac{2}{\delta n}.$$

So we can find w such that

$$\mathbb{E}_{v \in N(w)} \left[\frac{1}{d(v)} \right] + \frac{1}{d(w)} \leq \frac{2}{\delta n}.$$

The fact that the first term is small gives that most $v \in N(w)$ have large degree; the fact that the second is small says that $d(w) = \Omega(\delta n)$, so the neighborhood $N(w)$ is itself quite large.

So this gives the following fact:

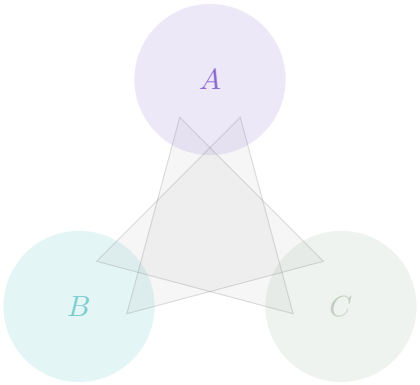
Lemma 1.2

Given any bipartite graph of density δ, we can find some $w \in B$ such that $|N(w)| = \Omega(\delta n)$, and most $v \in N(w)$ have degree $\deg(v) = \Omega(\delta n)$.

In words, we can find some vertex on the right with a large neighborhood, such that most vertices in this neighborhood have large degree.

§2 A generalization to 3-uniform hypergraphs

That's a warm-up; now we'll consider a more general version of this idea. Suppose that we have a 3-partite 3-uniform hypergraph — so we have vertex sets A, B, and C of size $|A| = |B| = |C| = n$, and G consists of a bunch of triangles between A, B, and C. We'll still say the edge density of G is δ, so $|E(G)| = \delta n^3$.



Question 2.1. What's the appropriate generalization of Lemma 1.2?

There are a few complexities with trying to come up with such a generalization. First, if we have a hypergraph, there are multiple ways to generalize the notion of degrees.

- If we have one vertex v, we can look at how many edges are incident to v; we call this $d(v)$.
- But there's also the notion of *codegrees*: if we have *two* vertices v and w, we can look at how many edges are incident to both v and w, which we call $d(v, w)$.

We'll mostly care about codegrees. The reason for this is that codegrees are associated to neighborhoods — we can define

$$N(v, w) = \{u \mid uvw \text{ is an edge}\},$$

so that $d(v, w) = |N(v, w)|$. On the other hand, if we look at a single vertex, the notion of its neighborhood is a bit sketchier and not really useful for our purposes.

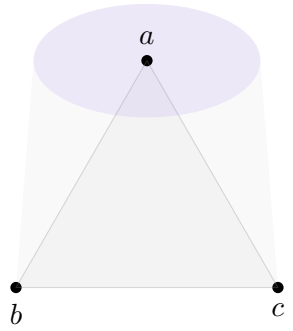
So our goal is to find large sets of pairs (v, w) where we have $d(v, w) = \Omega(\delta n)$. Even knowing this, it's still unclear what the correct generalization is. So we'll try to start proving things and see what falls out.

§2.1 A first attempt — induction

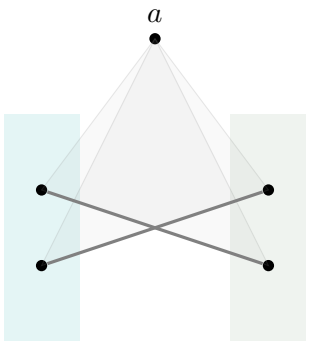
We'll start with a proof technique that seems natural, and later we'll see how to do slightly better. The main idea of this attempt is that we want to reduce back to the 2-uniform case.

The first observation is that whenever we have a hypergraph of this form, we can equivalently think of it as a bipartite *graph* where one vertex set is A and the other is $B \times C$ (since edges are ordered triples (a, b, c)

in either setting). If we apply Lemma 1.2 to this graph, we get that there exist $b \in B$ and $c \in C$ such that $d(b,c) = \Omega(\delta n)$, and for most $a \in N(b,c)$, we have $d(a) = \Omega(\delta n^2)$. (What we're using here isn't exactly the same as Lemma 1.2, because A has size n and $B \times C$ has size n^2, but it can be proven in the same way.)

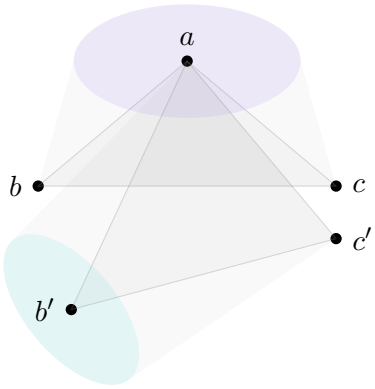


Then for each a with this property, we can restrict to this a and consider the bipartite graph on B and C that consists of the edges containing a (i.e., the bipartite graph with an edge bc whenever abc was an edge in the original hypergraph).



By Lemma 1.2 again, for each such a, there exists some $c' \in C$ such that c' has large degree in this bipartite graph, and most of its neighbors b' also have large degree in this bipartite graph. In other words, this means $d(a,c') = \Omega(\delta n)$, and for most $b' \in N(a,c')$, we have $d(a,b') = \Omega(\delta n)$.

This is something that looks like a generalization of Lemma 1.2 — we've created large neighborhoods $N(b,c)$, and if you look inside that large neighborhood you can find large neighborhoods $N(a,c')$, and if you look inside those you have large degrees.



There's a bunch of quantifiers here; to summarize what's going on, our statement looks like

$$(\exists\, b)(\exists\, c)(\forall\, a)(\exists\, c')(\forall\, b')$$

(really instead of $\forall$ we have 'most,' but that's not important). This is maybe a bit unsatisfying — we're choosing one b and c for everything, but we have to choose a separate c' for every single a, and that's a lot of information to carry around, which might seem a bit inelegant.

One nice thing about the 2-uniform case (Lemma 1.2) was that all the existential quantifiers were at the very beginning of the statement. So we can ask, is it possible to do the same thing here?

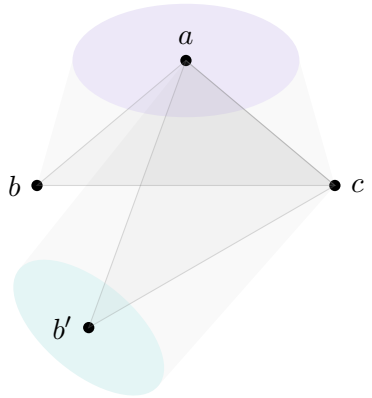
Question 2.2. Is it possible to move the quantifier order to $(\exists b)(\exists c)(\exists c')(\forall a)(\forall b')$?

In other words, our current proof has shown that there exist b and c such that for all (or rather, most) $a \in N(b,c)$, there exists c' such that certain things are true. But there's many choices of a, so there's a lot of information that comes with picking a separate c' for each. So our question is, is it possible to strengthen our statement so that we're using the *same* c' for every choice of a?

It turns out that the answer is yes, and the proof is very nice.

§2.2 Swapping the quantifiers

The key insight is that in fact, the statement we want is true even if we set $c' = c$.



The proof is going to use the following random process: first, we sample an edge $abc \in E(G)$ uniformly at random. By the same argument as in the 2-uniform case, we have

$$\mathbb{E}\left[\frac{1}{d(a,b)}\right] = \mathbb{E}\left[\frac{1}{d(b,c)}\right] = \mathbb{E}\left[\frac{1}{d(a,c)}\right] = \frac{1}{\delta n}.$$

Now suppose that we resample b as b' — this means we're choosing $b' \in N(a,c)$ uniformly at random. Then the distribution of $ab'c$ is still that of a uniform random edge (we're starting with the uniform distribution and resampling one coordinate, so the result is still uniform), so it's still true that

$$\mathbb{E}\left[\frac{1}{d(a,b')}\right] = \mathbb{E}\left[\frac{1}{d(b',c)}\right] = \frac{1}{\delta n}.$$

How does this help us? Now we can sort of mimic the 2-uniform proof — we start with the fact that

$$\mathbb{E}\left[\frac{1}{d(b,c)} + \frac{1}{d(a,c)} + \frac{1}{d(a,b')}\right] = \frac{3}{\delta n}.$$

This tells us that there's some b and c such that the expectation, if we condition on having these particular b and c, is at most $\frac{3}{\delta n}$, i.e.,

$$\mathbb{E}\left[\frac{1}{d(b,c)} + \frac{1}{d(a,c)} + \frac{1}{d(a,b')} \;\middle|\; b, c\right] = \frac{3}{\delta n}.$$

This tells us two things. First, it tells us that

$$\frac{1}{d(b,c)} \le \frac{3}{\delta n},$$

which means that $d(b,c) = \Omega(\delta n)$. Meanwhile, it also tells us that

$$\mathbb{E}\left[\frac{1}{d(a,c)} + \frac{1}{d(a,b')} \,\middle|\, b,c\right] \le \frac{3}{\delta n}.$$

By Markov's inequality, this means that for most a, we'll have

$$\mathbb{E}\left[\frac{1}{d(a,c)} + \frac{1}{d(a,b')} \,\middle|\, a,b,c\right] = O\left(\frac{1}{\delta n}\right).$$

The first term being small means that $d(a,c) = \Omega(\delta n)$ is large; the second term being small means that

$$\mathbb{E}_{b'}\left[\frac{1}{d(a,b')}\right] = O\left(\frac{1}{\delta n}\right)$$

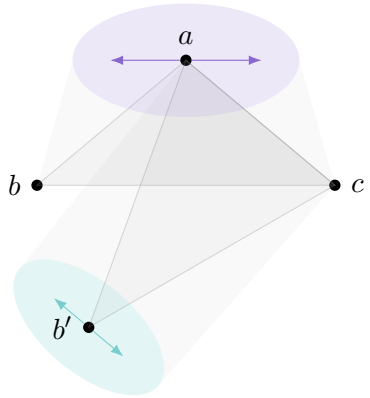
is small, so $d(a,b') = \Omega(\delta n)$ for most b'.

That was a lot of stuff, but to summarize, we've shown that you can find some b and c such that there are lots of choices for $a \in N(b,c)$; and for most such choices of a, there are lots of choices for $b' \in N(a,c)$; and $d(a,b')$ is large for most of these choices:

Lemma 2.3

Given any 3-partite 3-uniform hypergraph of edge density δ, we can find $b \in B$ and $c \in C$ such that:

- We have $|N(b,c)| = \Omega(\delta n)$.
- For most $a \in N(b,c)$, we have $|N(a,c)| = \Omega(\delta n)$, and most $b' \in N(a,c)$ satisfy $d(a,b') = \Omega(\delta n)$.

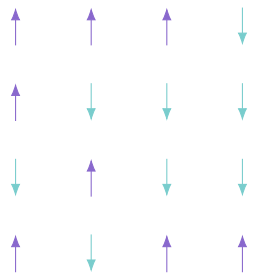


(The arrows indicate that a and b' are allowed to move around.)

§3 Connection to statistical physics

Let's talk philosophically about what happened here. The key idea was resampling — if we choose an edge uniformly at random and resample one of its vertices, it's still a uniform random edge. In some sense, we've essentially treated the graph as some sort of statistical ensemble. To see what this means, we'll draw some comparisons to the Ising model from statistical physics.

The idea of the Ising model is that it's a model of magnetism inside some solid-state body, where we have a bunch of spins that are either up or down.



Then we can run something called the *Glauber dynamics*, which is an algorithm for sampling from the Ising model. The idea is that we start from any configuration and then take a random walk on the configuration space where at each step, we choose one element uniformly at random and resample it (according to some rule that only depends on its neighbors).

You can show that this algorithm mixes all the states of the model, and it converges to the right distribution.

In our setting, we have a hypergraph, and we're taking a random edge and randomly resampling its vertices; this is something that preserves the uniform distribution on edges. We're starting with some edge, and the resampling steps cause us to take a random walk through the edges of the hypergraph. (In the 3-uniform setting we only had one resampling step; but if we replaced the uniformity 3 with d, then we'd have $d-2$ resamplings — we resample every vertex except the first and last.)

We can think of the hypergraph as an ensemble of edges, and each edge as a configuration; so we're taking a random walk through the set of configurations.

And similarly to the Glauber dynamics on the Ising model, the point is that this random walk allows us to relate *global* densities to *local* behavior — here, we have a global parameter δ telling us how many edges are in the graph, and we're able to relate it to the codegrees $d(u,v)$ (which are kind of local information).

§4 Local rank

That's the combinatorial principle that Daniel wanted to talk about; we'll spend the remaining time discussing the problem that they developed this technique for.

Let $\mathbb{F} = \mathbb{F}_q$ be a finite field. We'll work with a multlinear map $T\colon (\mathbb{F}^n)^d \to \mathbb{F}^n$ — this means T takes in d vectors in $\mathbb{F}^n$, and outputs another vector in $\mathbb{F}^n$.

Example 4.1

If $n=2$ and $d=3$, then we could have something like

$$T((a_1,a_2),(b_1,b_2),(c_1,c_2)) = (a_1b_1c_2 + 2a_2b_2c_2, a_1b_2c_2).$$

We use $\mathcal{Z}(T)$ to denote the zero set of T, i.e., $\mathcal{Z}(T) = T^{-1}(0)$.

Question 4.2. If $\mathcal{Z}(T)$ is large, what can we say about T?

To get a sense of what a correct answer to this problem could possibly look like, here's an observation: if all the components of T have a common factor, then whenever that common factor is 0, of course T will also output 0. And one can show that such a common factor has to be linear, so it'll be 0 for at least $\frac{1}{q}$ of all inputs. So in this case, we'll have

$$\frac{|\mathcal{Z}(T)|}{q^{nd}} \geq \frac{1}{q}$$

(where q^{nd} is the total number of possible inputs).

Definition 4.3. We call T reducible if all its components have a common factor.

The following more general fact is also true, though it's harder to see.

Proposition 4.4

If T is the sum of r reducible maps, then

$$\frac{|\mathcal{Z}(T)|}{q^{nd}} \geq \frac{1}{q^r}.$$

So the kind of statement we want to show is that if

$$\frac{|\mathcal{Z}(T)|}{q^{nd}} \geq \frac{1}{q^a}$$

(where a is small — this corresponds to $\mathcal{Z}(T)$ being large), then T is the sum of r reducible maps. And the main question we're interested in is:

Question 4.5. What is the dependence of r on a?

First, you can show that such a function does exist in the first place — i.e., there is some $r = f_d(a)$ for which this is true. This comes from a result of Green–Tao (2009), and some refinements by a few others.

More recently, it's been shown that in fact r is bounded *polynomially* in a — i.e., we can take $r = O_d(a^{c_d})$. This was proven independently by Milićević (2019) and Janzer (2020).

Conjecture 4.6 — We can take $r = O_d(a)$.

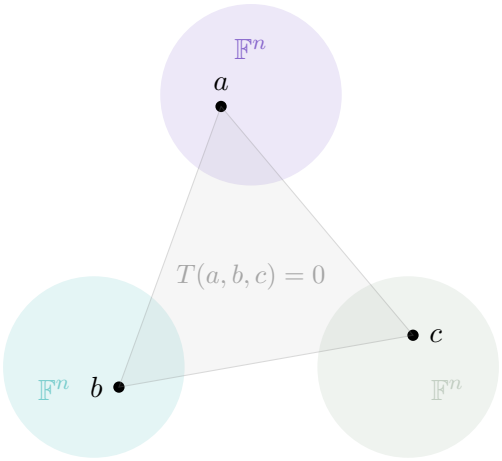
In the regime where q is very large in terms of a, this was shown by Alex Cohen and Guy Moshkovitz a few years ago. This is also known in the case $d = 2$ (the $d = 1$ case is kind of trivial, and the $d = 2$ case is the smallest nontrivial case; and there we do know this conjecture is true).

Theorem 4.7 (Moshkovitz–Zhu)

We can take $r = O_d(a \log_q a)$.

(The original proof got a slightly worse bound, but since then there have been some minor improvements to the end of the proof, ending up with this bound.)

Daniel will finish by talking about how the earlier principle ends up applying to this problem. That principle involved a hypergraph, so here we need to construct one. We'll draw the d-uniform hypergraph where the vertex sets are $\mathbb{F}^n$, and every edge corresponds to an element of $\mathcal{Z}(T)$. We'll assume $d = 3$ for concreteness; then we draw an edge between three vertices if plugging those three things into T gives 0.



Now applying Lemma 2.3 gives us some information about codegrees. Here, the codegrees are given by

$$d(a,b) = \#\{c \mid T(a,b,c) = 0\}.$$

On the other hand, since T is multilinear, the map $T(a,b,c)$ is linear as a function of c. This means if we normalize by q^n, then $d(a,b)$ is completely controlled by the *rank* of this linear map — we have

$$\frac{d(a,b)}{q^n} = q^{-\operatorname{rank} T(a,b,\bullet)}.$$

Then Lemma 2.3 tells us that given information about the global density (i.e., the size of $\mathcal{Z}(T)$), we can construct a large structure where most of the codegrees are large; and this means when we plug entries into all but one of the arguments of T, most of the ranks we get will be small.

Then the rest of the proof can be summarized as two steps. The first step is a sort of *rigidity* argument. The statement we get from Lemma 2.3 has a lot of 'most's. What rigidity does is that it sort of turns the 'most's into 'all's (there's an asterisk here, but this is morally true in some sense).

Then there's a *structure-building* step. What this does is that if we have an 'all' statement (with 100% instead of 90%), then it allows us to construct a decomposition into a sum of r reducible maps. (There's also a very serious asterisk here that we're not going to talk about.)

(Both of these steps can be considered applications of the polynomial method — they involve lots of algebraic structure, and you can write down polynomials that deal with the relevant conditions. Once you've constructed the polynomial, the rigidity step ends up amounting to the fact that if you have a polynomial which vanishes at most points in a vector space, then it has to vanish at all of them.)

So to recap, the key step in the proof was starting with this sort of arbitrary structure $\mathcal{Z}(T)$, and finding a way to output some structured part of the hypergraph where all codegrees are large. And in that case, you can sort of apply linear algebra, and it ends up giving you the structure result we wanted.

> **Remark 4.8.** The place where the $\log a$ term comes from is that part of the asterisks is that we can only get these two steps to work out if q is at least some polynomial in a. This isn't always true; so to get around this, we pass to a large field extension, run the argument there, and pass back down. And this translation is where you lose the log factor.