Recall the definition of probabilistically checkable proofs (PCP) from last time. We say L has a PCP of randomness complexity $r(n)$ and query complexity q if there is a randomized polynomial-time verifier V which, on input x of length n, and oracle access to a proof $\pi \in \{0,1\}^*$, uses at most $r(n)$ random coins to select q nonadaptive queries into π such that

$$x \in L \longrightarrow \exists \pi : \Pr[V^\pi(x) = 1] = 1$$
$$x \notin L \longrightarrow \forall \pi : \Pr[V^\pi(x) = 1] \leq 1/2.$$

Today we will prove that every NP problem has a PCP of randomness complexity $O(\log n)$ and constant query complexity.

It will be useful to pay attention to two more parameters of the PCP which we fixed to constants in the above definition. One is the *soundness*, which is $1/2$ above. Another one is the *proof alphabet size*. In our definition of PCP we assumed that the when the verifier queries an entry in the proof, the answer is a single bit. It will be helpful to generalize to answers that consist of a symbol from some larger alphabet Σ. We get the following definition:

Definition 1. We say L has a PCP of randomness complexity $r(n)$, query complexity q, full completeness, soundness $s(n)$, and proof alphabet Σ if there is a randomized polynomial-time verifier V which, on input x of length n, and oracle access to a proof $\pi \in \Sigma^*$, uses at most $r(n)$ random coins to select q nonadaptive queries into π such that

$$x \in L \longrightarrow \exists \pi : \Pr[V^\pi(x) = 1] = 1$$
$$x \notin L \longrightarrow \forall \pi : \Pr[V^\pi(x) = 1] \leq s(n).$$

Let's write this as $L \in \mathrm{PCP}^{\Sigma}_{1,s(n)}(r(n), q)$.

For the proof of the PCP theorem, it will be helpful to shift between the proof system and constraint satisfaction views of PCPs. In this more general setting, the statement

> L has a PCP with randomness complexity $r(n) = O(\log n)$, query complexity q, full completeness, soundness $s(n)$, and proof alphabet Σ

is equivalent to

> L reduces to promise problem whose instances are systems of $2^{r(n)}$ constraints over alphabet Σ with q variables with constraint, where in the yes instances all constraints are simultaneously satisfiable, and in the no instances no assignment satisfies more than an $s(n)$ fraction of constraints.

1 Irit Dinur's proof of the PCP theorem

The starting point of Dinur's proof of the PCP theorem is the fact that the NP-hardness of 3SAT, can be viewed as an extremely weak hardness of approximation result. For a 3SAT instance with n variables, the fact that 3SAT is NP-hard can be restated as saying that it is NP-hard to distinguish satisfiable 3CNFs from those 3CNFs where at most $1 - 1/m$-fraction constraints can be satisfied, where $m \leq 8\binom{n}{3}$ is the number of constraints. The idea is to start from this statement and design a sequence of transformations that gradually improve the soundness parameter at a very small cost in the randomness complexity, while leaving all the other parameters unchanged.

Let us now view 3SAT as a qCSP, i.e. a CSP with q variables per constraint (in this case $q = 3$) and alphabet $\{0, 1\}$. Dinur designs a polynomial-time transformation with the following effect:

	Ψ (qCSP instance)	$\to \Psi'$ (qCSP instance)
number of edges	m	$\to Cm$ (C is some constant)
completeness	1	$\to 1$
soundness	$1 - \delta$	$\to 1 - 2\delta$ (for $\delta < 1/C$)

Repeating this transformation $O(\log m)$ times yields the PCP theorem.

The transformation consists of several stages (smaller transformations), where at each stage one of the parameters is improved, but at the expense of the others. At the end, we achieve an improvement in soundness while paying a small price in the size of the instance (randomness complexity of the PCP).

To explain the various parameters, we need a bit of notation. The *constraint graph* of a CSP Ψ with 2 variables per constraint (equivalently a PCP with query complexity 2) is the graph whose vertices v correspond to variables x_v in the CSP and where for each constraint $\psi(x_u, x_v)$ in Ψ there is an edge (u, v) in the graph. Multiple edges are allowed, and we will keep the graph undirected (as the CSP constraints will be symmetric).

The *size* of a CSP is the number of constraints in it. We will say Ψ has *degree* d if its constraint graph is a regular graph of degree d, and the *expansion* of Ψ is the spectral gap $1 - \lambda(G)$ of its underlying constraint graph G.

Dinur's proof of the PCP theorem goes through the following sequence of transformations:

transformation	size	$\lvert\Sigma\rvert$	queries	soundness gap	degree	spectral gap
	m	2	$q = C$	δ		
reduce queries	$\times C$	2^q	2	$\div C$	large	
reduce degree	$\times C$	2^q	2	$\div C$	C	small
expanderize	$\times C$	2^q	2	$\div 2$	$d = 2C$	1/4
amplify gap	$\times d^t$	2^{qd^t}	2	$\times t/\lvert\Sigma\rvert^4$		
reduce alphabet	$\times \exp(2^{qd^t})$	2	q	$\div 2$		

The only quantities that are non-constant throughout this sequence of transformations are the size (initially m) and the soundness gap (initially δ). When the parameters are chosen appropriately (t

is a sufficiently large constant in terms of C), the soundness gap is doubled, while the size of the instance increases only by a constant factor, giving the desired conclusion.

2 PCP transformations

We now describe the five PCP transformations in the proof of the PCP theorem. We will describe query reduction, degree reduction, and expanderizing in this section. Gap amplification and alphabet size reductions are described later on.

2.1 Query reduction

The goal of this transformation is to take a q-query PCP (for some constant q) and turn it into a 2-query PCP. Think of the q-query PCP as a constraint satifaction problem Ψ with n variables $x_1, \ldots, x_n$ and m constraints over alphabet $\{0,1\}$. We define a 2-query CSP Ψ' as follows:

- **Variables of Ψ':** The instance Ψ' has variables $x_1, \ldots, x_n, y_1, \ldots, y_m$, where x_i takes values in $\{0,1\}$ and y_i takes values in $\Sigma = \{0,1\}^q$. Intuitively, the value of y_i should be $x_{i_1} \ldots x_{i_q}$, where $x_{i_1}, \ldots, x_{i_q}$ are the variables participating in the constraint ψ_i of Ψ.
- **Constraint graph of Ψ':** For each constraint ψ_i of Ψ and each variable x_{i_j} that participates in ψ_i, there is a constraint-edge (x_{i_j}, y_i). The corresponding constraint is satisfied if the jth entry in y_i equals x_{i_j}.

Clearly, if Ψ is satisfiable, so is Ψ' (we just assign $y_i = x_{i_1} \ldots x_{i_q}$ for every i). On the other hand, if every assignment violates a δ-fraction of constraint in Ψ, then every assignment will violate a δ/q-fraction of constraints in Ψ'. To prove this, assume that some assignment (x, y) violates less than a δ/q-fraction of constraints in Ψ'. Since every y_i is involved in q constraints, it means that all constraints involving y_i are satisfied for at least a $1 - \delta$ fraction of y_is. But if all constraints involving y_i are satisfied, it must be that x satisfies ψ_i in Ψ, so x satisfies a $1 - \delta$ fraction of constraints in Ψ.

2.2 Degree reduction

In general, after applying query reduction, the constraint graph may have very large degree. The goal of the degree reduction step is to make the degree constant (independent of the instance size), while losing only a constant factor in the soundness gap.

This transformation makes use of expanders. Recall that a d-regular graph has *edge expansion* $0 \leq h \leq 1$ if for every set S that contains no more than half the vertices, the number of edges between S and $\overline{S}$ is at least $\alpha|S|$. Using the zig-zag product we showed how to construct expanders with $h = 1/4$ and constant degree d.

Let G be the constraint graph of Ψ. We create a new CSP Ψ' by replacing every vertex i in G of degree n_i by a cloud of n_i vertices. So each variable x_i of Ψ will give rise to d_i variables $x'_{i1}, \ldots, x'_{in_i}$

in Ψ'. Each constraint in Ψ gives rise to $d/2$ parallel constraints in Ψ' between unique vertices in the corresponding clouds. Within each cloud, we interconnect the vertices by a 1/4-edge expander and make each expander constraint an equality constraint (i.e. requiring that variables get the same value). Notice that if Ψ has m constraints, then Ψ' will have m variables and dm constraints.

Clearly, if Ψ has a satisfying assignment x, we can obtain a satisfying assignment for Ψ' by setting $x'_{i1} = \cdots = x'_{in_i} = x_i$ for every i.

Now we prove soundness. The fact that the soundness gap goes down by at most a constant factor in this transformation is a consequence of the following claim:

Claim 2. *If some assignment x' the violates at most an ε-fraction of contraints in Ψ', then there exists an assignment x that violates at most a 18ε fraction of constraints in Ψ.*

The assignment x is obtained from x' as follows: Within each cloud, let x_i be the plurality value (i.e., the most representative value) among $x'_{i1}, \ldots, x'_{in_i}$. Let ε_i be the fraction of constraints violated in cloud i. Then $\sum_{i=1}^{n} \varepsilon_i \cdot (dn_i/4) \leq \varepsilon \cdot (dm/2)$, the total number of violated constraints.

Let's fix a cloud i. Let S_i be the set of vertices j within this cloud where x'_{ij} agrees with the plurality assignment. We will argue that, because of the expansion in the cloud, the assignment within the cloud must largely agree with the plurality assignment. We split the analysis into three cases:

- If $|S_i| > n_i/2$, then by edge expansion $|E(S_i, \overline{S}_i)| \geq d|\overline{S_i}|/4$. Since all the constraints in the cut $(S_i, \overline{S}_i)$ are violated by the assignment, $|E(S_i, \overline{S}_i)| \leq \varepsilon_i(dn_i/4)$, so $|\overline{S_i}| \leq 4\varepsilon_i n_i$.
- If $n_i/4 \leq |S_i| \leq n_i/2$, then by edge expansion $|E(S_i, \overline{S}_i)| \geq d|S_i|/4 \geq dn_i/16$. Since all the constraints in the cut are violated, it follows that $\varepsilon_i \geq 1/4$, so $|\overline{S_i}| \leq n_i \leq 4\varepsilon_i n_i$.
- If $|S_i| < n_i/4$, then no value in Σ is taken more than 1/4-fraction of the time inside the cloud, so there must exist some partition of the values within the cloud so that the smaller side of the partition has between $n_i/4$ and $n_i/2$ vertices. Just like in the previous case, we get that $|\overline{S_i}| \leq n_i \leq 4\varepsilon_i n_i$.

We see that no matter what, $|\overline{S_i}| \leq 4\varepsilon_i n_i$ for every i.

Now consider what happens in Ψ' when we replace the assignment x' with the plurality assignment x'_{plur} (i.e. one that equals the plurality of x' on every cloud). For each cloud, this may cause the violation of at most $(d/2)|\overline{S_i}|$ additional constraints that go out of the cloud. So if x' violates εdm constraints in Ψ', x'_{plur} will violate at most

$$\varepsilon(dm/2) + \sum_{i=1}^{n}(d/2)|\overline{S_i}| \leq \varepsilon(dm/2) + \sum_{i=1}^{n}(d/2)(4\varepsilon_i n_i) \leq \varepsilon(dm/2) + 8\varepsilon(dm/2) = 9\varepsilon(dm/2)$$

constraints of Ψ'. This is a 9ε-fraction of all the constraints in Ψ'. So the assignment x can violate at most a 18ε fraction of constraints in Ψ.

2.3 Expanderizing

The expanderizing transformation starts with a CSP Ψ with two variables per constraint and (sufficiently large) constant degree d and creates a new CSP Ψ' with two variables per constraint, degree $2d$, and the property that the constraint graph is an expander with $\lambda \leq 3/4$.

This transformation is very simple. Suppose the constraint graph G of Ψ has n vertices. Let Z be an expander on n vertices with degree d and $\lambda \leq 1/2$ (which we can construct in polynomial time using the zig-zagging method). The variables of Ψ' will be the same as the variables of Ψ. The constraints of Ψ' will incude all the constraints of Ψ. In addition, for every edge in Z, we add a "dummy" constraint in Ψ', namely one which is satisfied for any assignment to its endpoints.

Clearly if Ψ is satisfiable, then Ψ' is satisfiable by the same assignment. On the other hand, if x fails to satisfy a δ fraction of the constraints in Ψ, then it will fail to satisfy the same constraints in Ψ', which form a $\delta/2$-fraction of all the constraints.

Let G' be the constraint graph of Ψ'. We now show that if G and Z are regular graph of the same degree, then $\lambda_{G'} \leq (\lambda_G + \lambda_Z)/2 \leq 3/4$. Notice that the adjacency matrices satisfy the relation $A_{G'} = (A_G + A_Z)/2$. Then for every $\mathbf{v} \perp \mathbf{u}$, we have

$$\|\mathbf{v}A_{G'}\| \leq \tfrac{1}{2}\big(\|\mathbf{v}A_G\| + \|\mathbf{v}A_Z\|\big) \leq \tfrac{1}{2}\big(\lambda_G\|\mathbf{v}\| + \lambda_Z\|\mathbf{v}\|\big) = \tfrac{1}{2}(\lambda_G + \lambda_Z)\|\mathbf{v}\|.$$

3 Gap amplification

Fix a constant t. The gap amplification step is a transformation from a 2CSP Ψ with degree d and $\lambda = 3/4$ to a 2CSP Ψ' with the following parameters:

	Ψ	$\to \Psi'$
size	m	$\to (\lvert\Sigma\rvert d)^{5t} m$
alphabet size	Σ	$\to \Sigma^{1+d+d^2+\cdots+d^t}$
completeness	1	$\to 1$
soundness	$1-\delta$	$\to 1 - \Omega(t\delta/\lvert\Sigma\rvert^4)$

We will describe a slightly modified version of Dinur's transformation due to Jaikumar Radhakrishnan. Let G be the constraint graph of Ψ. We define the instance Ψ' as follows:

- **Variables of Ψ':** For each variable x_v of Ψ, there is a corresponding variable x'_v of Ψ'.
- **Values of x'_v:** The value of x'_v is a collection (tuple) of values in Σ, one corresponding to every vertex u at distance $\leq t$ from v in G. We write $x'_v(u)$ for the component of x'_v corresponding to u.
- **Distribution over constraints of Ψ':** The constraints ψ'_p of Ψ' correspond to paths p of length at most $5t \ln|\Sigma|$ in G. (We will identify constraints and the paths they represent.) The paths are generated from the following distribution:

1. Choose a starting vertex v_0 of p. Set $i = 0$
2. Repeat for at most $5t \ln|\Sigma|$ times: (1) Set v_{i+1} to be a random neighbor of v_i and increment i (2) With probability $1/t$, stop the repetition.
3. Output the path $v_0, v_1, \ldots, v_i$.

- **Constraints of Ψ':** Let (u', v') be the endpoints of a path p. The constraint $\psi'_p(x'_{u'}, x'_{v'})$ is satisfied if all of the following hold:
  1. For every edge (u, v) in G such that u and v are both within distance t of u', the constraint $\psi_{(u,v)}$ is satisfied.
  2. For every edge (u, v) in G such that u and v are both within distance t of v', the constraint $\psi_{(u,v)}$ is satisfied.
  3. For every vertex v that is within distance t from both u' and v', $x'_{u'}(v) = x'_{v'}(v)$.

The size and alphabet size of Ψ' are easy to check. We need to argue completeness and soundness. By design, the transformation has perfect completeness. Suppose x is a satisfying assignment of Ψ. Now consider the assignment x' of Ψ' given by $x'_u(v) = x_v$. This satisfies all the constraints of Ψ'.

The (relatively) difficult part is to argue soundness. To do this, we must show that for every x' that satisfies $1 - \Omega_\Sigma(t\delta)$ constraints of Ψ', there is an x that satisfies $1 - \delta$ constraints of Ψ.

The assignment x is constructed from x' via the following procedure. For every vertex v,

1. Define the following distribution D_v on vertices. Initially, set $v' = v$. Now repeat the following experiment: With probability $1/t$ stop, and with the remaining probability, set $v' =$ a random neighbor of v'.
2. Set x_v to equal the plurality value of $x'_{v'}(v)$, when v' is chosen from D_v, among those v' that are within distance t of v.

We now need to argue that if x' satisfies $1 - \Omega(t\delta/|\Sigma|^4)$ constraints of Ψ', then x satisfies $1 - \delta$ constraints of Ψ. In fact, we will argue the contrapositive:

Claim 3. *Assume $t\delta < 1$. If x violates δ constraints of Ψ, then x' violates $\Omega(t\delta/|\Sigma|^4)$ constraints of Ψ'.*

Before we prove the claim, let us make one simplification. We will modify the distribution over constraints of Ψ' so that the path p is not truncated after $5t \ln|\Sigma|$ steps (see step 2), but can be of any length. Intuitively, this simplification should not make a difference because long paths are unlikely. Formally, we will analyze the effect of this simplification later.

Now let's explain the intuition behind this claim. Let F be the set of constraints of Ψ (which we also think of as edges of G) that are violated by x (so $|F| = \delta m$). Now take a random constraint ψ' of Ψ'. What are the chances that this constraint is violated by x'? We have that

$$\Pr[x' \text{ violates } \psi'] = \Pr[\psi' \text{ intersects } F] \cdot \Pr[x' \text{ violates } \psi' \mid \psi' \text{ intersects } F].$$

Let's try to estimate both of these quantities. We expect ψ' to have about t edges; since $|F| = \delta m$, we expect ψ' to contain about δt edges of F. Since δ is fairly small, we might expect that most ψ' which intersect F intersect only a single edge of F. If this is the case, then $\Pr[\psi' \text{ intersects } F]$ should be about δt. Roughly, **this is where the soundness amplification happens**: While "bad" edges occur in Ψ only with probability δ, they occur in Ψ' with probability about $t\delta$.

What about the other probability? Let's now fix an edge $(u, v) \in F$ that is contained in ψ'. Now consider the distribution of the endpoints u' and v' of the path ψ'. Since the endpoints of the path are determined by a Poisson process, it follows that conditioned on (u, v) being in ψ', the endpoint v' is determined by the following distribution: Start from v and at each step (1) with probability $1/t$ stop and (2) with the remaining probability move to a random neighbor of v and continue. But this is exactly the distribution D_v! Ignoring for now the fact that ψ' could be too long, we reason as follows. Since the value x_v was defined as the plurality value $x_{v'}(v)$, the two should match with probability at least $1/|\Sigma|$. For the same reason, $x_{u'}(u)$ and x_u should match with probability $1/|\Sigma|$. But since the constraint $\psi(x_u, x_v)$ is violated, this implies $\psi'(x'_{u'}, x'_{v'})$ is also violated.

So roughly, we expect that the probability that a random constraint of Ψ' is violated is about $\delta t/|\Sigma|^2$. However, our "analysis" ignored several crucial points, namely:

- Why can we assume that few ψ' intersect multiple edges of F?
- What happens when ψ' contains more than t edges? In this case, it may happen that ψ' contains a "bad" edge, but this edge cannot be "seen" from its endpoints.

Roughly, the answer is: (1) The fact that ψ' is unlikely to intersect many edges of F follows from the expansion of G and (2) Long paths will contribute little to the analysis as they only happen with small probability.

3.1 Analysis of gap amplification

Now let us do the actual analysis. Call an edge (u, v) *faulty* (with respect to ψ', x', x) if (1) $(u, v) \in F$, (2) $d(u', u), d(v, v') < t$, and (3) $x'_{u'}(u) = x_u$ and $x'_{v'}(v) = x_v$, where u', v' are the endpoints of ψ'. If some edge in ψ' is faulty, then ψ' is violated as the inconsistency between x_u and x_v can be seen either by $x'_{u'}$ or by $x'_{v'}$.

Let N denote the number of faulty edges of ψ', where ψ' is chosen at random. We have that

$$\Pr[\psi' \text{ is violated}] \geq \Pr[N > 0] \geq \mathrm{E}[N]^2 / \mathrm{E}[N^2]. \tag{1}$$

The first moment. We first estimate $\mathrm{E}[N]$. For $f \in F$, let I_f denote the number of occurrences of f in ψ', and let $N_f = I_f$ is f is faulty, and 0 otherwise. Then:

$$\mathrm{E}[N] = \sum_{f \in F} \mathrm{E}[N_f] = \sum_{f \in F} \sum_{k=1}^{\infty} \Pr[N_f \geq k] = \sum_{f \in F} \sum_{k=1}^{\infty} k \cdot \Pr[I_f \geq k] \cdot \Pr[f \text{ is faulty} \mid I_f \geq k].$$

Let us analyze the probability that f is faulty conditioned on $I_f \geq k > 0$. Fix an arbitrary collection of k occurrences of f in ψ and let u be the left endpoint of the first occurrence and v

be the right endpoint of the last occurrence. As discussed above, u' follows the distribution D_u, and v' independently follows the distribution D_v. In this distribution, the probability that u' is at distance more than t from u is $\leq (1-1/t)^t < 1/2$. Conditioned on this distance being at most t, the distribution on u' is exactly the one used to define the plurality assignment x_u, so the probability that $x'_{u'}(u) = x_u$ is at least $1/|\Sigma|$. As the same is true for v and v' independently, for any $k > 0$ we have that

$$\Pr[f \text{ is faulty} \mid I_f \geq k] \geq \left(\frac{1}{2} \cdot \frac{1}{|\Sigma|}\right)^2$$

and therefore

$$\mathrm{E}[N] \geq \frac{1}{4|\Sigma|^2} \cdot \sum_{f \in F} \sum_{k=1}^{\infty} \Pr[I_f \geq k] = \frac{1}{4|\Sigma|^2} \cdot \sum_{f \in F} \mathrm{E}[I_f] = \frac{\delta t}{4|\Sigma|^2},$$

because the expected number of occurrences of any particular edge in ψ' is $1/m$ times the expected length of ψ', which is t.

The second moment. We now upper bound $\mathrm{E}[N^2]$. To do so, let N' be the number of edges in F that intersect ψ'. Obviously $N \leq N'$ (since N counts the number of such edges that are also faulty). So we will bound $\mathrm{E}[N'^2]$ instead. To do so, let Z_i be a random variable that indicates if the ith edge of ψ' is in F (if ψ' has fewer than i edges, then $Z_i = 0$). Then

$$\mathrm{E}[N'^2] = \sum_{i=1}^{\infty} \mathrm{E}[Z_i] + 2 \sum_{1 \leq i < j} \mathrm{E}[Z_i Z_j]. \tag{2}$$

It is easily seen that $\mathrm{E}[Z_i] = \delta \cdot (1 - 1/t)^i$, so the first summation is at most $t\delta$.

For the second summation, notice that $\mathrm{E}[Z_i Z_j]$ is the probability that both edges i and j are present in the path and faulty. The probability they are both present is $(1 - 1/t)^j$. Conditioned on them being both present, the probability they are both faulty is bounded using the following lemma.

Lemma 4. *Let G be a d-regular graph with spectral gap $1 - \lambda$ and F be a subset consisting of a δ fraction of the edges of G. The probability that both the first and the last edge of a random walk of G of length $\ell \geq 2$ are in F is at most $\delta^2 + \delta\lambda^{\ell-2}$.*

It follows that $\mathrm{E}[Z_i Z_j] \leq (1 - 1/t)^j \cdot \delta \cdot (\delta + \lambda^{j-i-1})$. Plugging this in (2) we have

$$\begin{aligned}
\mathrm{E}[N'^2] &\leq \delta t + 2\delta \sum_{1 \leq i < j} \mathrm{E}[Z_i Z_j] \\
&\leq \delta t + 2\delta \sum_{i=1}^{\infty} (1 - 1/t)^i \sum_{j=i+1}^{\infty} (1 - 1/t)^{j-i} \cdot (\delta + \lambda^{j-i-1}) \\
&\leq \delta t + 2\delta \sum_{i=1}^{\infty} (1 - 1/t)^i (\delta t + 1/(1 - \lambda)) \\
&\leq \delta t + 2\delta t(\delta t + 4) \\
&= 9\delta t + 2(\delta t)^2.
\end{aligned}$$

Second moment calculation. Finally, from (1) we have:

$$\Pr[N > 0] \geq \frac{\mathrm{E}[N]^2}{\mathrm{E}[N^2]} \geq \frac{(\delta t/4|\Sigma|)^2}{9\delta t + 2(\delta t)^2} = \Omega(\delta t/|\Sigma|^4).$$

The effect of truncation. This calculation was done in the idealized setting where ψ' can be arbitrarily long, while it is actually restricted to have length at most $5t\ln|\Sigma|$. It is not hard to see that these long paths contribute little to N. In particular, the contribution from the long paths can be bounded by

$$\sum_{\ell=5t\ln|\Sigma|}^{\infty} \mathrm{E}[N \mid \psi' \text{ has length } \ell] \Pr[\psi' \text{ has length } \ell] \leq \sum_{\ell=5t\ln|\Sigma|}^{\infty} (\delta\ell) \cdot (1-1/t)^{\ell} < \mathrm{E}[N]/2$$

For the calculation of $\mathrm{E}[N^2]$, the truncation of long paths only improves this quantity, so the lower bound on the probability that $N > 0$ is only affected by a constant.

Proof of Lemma 4. Let A be the (normalized) adjacency matrix of G and A' be the adjacency matrix of a graph representing $\ell - 2$ steps of a random walk on G. Then $A' = A^{\ell-2}$ and $\lambda' = \lambda^{\ell-2}$.

In Lecture 8 we proved that if we write $A' = (1-\lambda')J + E$ then for every vector $\mathbf{v}$, $\|\mathbf{v}E\| \leq \lambda'\|\mathbf{v}\|$. Here J represents the complete graph on n vertices. Now we write

$$\frac{1}{2n}|\mathbf{v}A'\mathbf{v}^{\mathrm{T}}| \leq (1-\lambda')\frac{1}{2n}|\mathbf{v}J\mathbf{v}^{\mathrm{T}}| + \frac{1}{2n}|\mathbf{v}E\mathbf{v}^{\mathrm{T}}|.$$

Let $\mathbf{v}$ be the vector such that $\mathbf{v}(u)$ equals the fraction of edges incident to u that are in F. Then $(\mathbf{v}A'\mathbf{v}^{\mathrm{T}})/2n$ equals exactly the fraction of paths with the first and last edge in F, and $(\mathbf{v}J\mathbf{v}^{\mathrm{T}})/2n$ equals $\mathrm{E}_u[\mathbf{v}(u)]^2 = \delta^2$. For the last term we have

$$\frac{1}{2n}|\mathbf{v}E\mathbf{v}^{\mathrm{T}}| \leq \frac{1}{2n}\|\mathbf{v}E\| \cdot \|\mathbf{v}\| \leq \lambda'\frac{1}{2n}\|\mathbf{v}\|^2 \leq \lambda'\frac{1}{2n}\sum_u \mathbf{v}(u) = \delta\lambda',$$

so the desired quantity is at most $(1-\lambda')\delta^2 + \delta\lambda' \leq \delta^2 + \delta\lambda^{\ell-2}$. □

4 Alphabet size reduction

The purpose of alphabet size reduction is to transform a 2-query PCP Ψ over large (but constant) alphabet Σ into a q-query PCP Ψ' over alphabet $\{0,1\}$, where q is independent of the size of Σ. We want to preserve completeness and lose only a constant factor (independent of Σ) in the soundness gap. On the other hand, the size of the instance is allowed to increase by a constant factor, which may depend on Σ.

Without loss of generality, we can think of Σ as $\{0,1\}^{\sigma}$ for some constant c. Then we can think of every variable y_i of Ψ as taking values in $\{0,1\}^{\sigma}$ and we can view every constraint $\psi(y_i, y_j)$ of Ψ as a function from $\{0,1\}^{2\sigma}$ to $\{0,1\}$.

Assume Ψ has full completeness and soundness δ. Now consider the following candidate PCP: The proof is a string of length $\{0,1\}^{\sigma n}$ which for every y_i contains all the bits of y_i. The verifier chooses a random constraint $\psi(y_i, y_j)$, reads the bits of y_i and y_j, and accepts if the constrant is satisfied. This is a PCP of full completeness and soundness δ. However, its query complexity is 2σ, which is too large: We want a PCP whose query complexity is independent of σ.

This is much like the situation we had in the last lecture: It seems that what we need is a PCP which proves that $\psi(y_i, y_j) = 1$ with query complexity independent of σ. Since the size of ψ itself is a most $O(2^{2\sigma}))$ (if we represent it as a circuit), the size of the PCP will only go up by a factor that depends on σ.

To implement the PCP construction from last lecture, we want to transform each constraint $\psi(y_i, y_j)$ into an "equivalent" system of quadratic equations Q. Recall that the system Q will have at most $O(2^{2\sigma}))$ equations; however in addition to the variables y_i and y_j, the system will also depend on at most $O(2^{2\sigma}))$ auxiliary variables z_{ij}.

The proof in the PCP Ψ' will now consist of two parts:

1. For each y_i taking values in $\{0,1\}^\sigma$, an encoding $C_i \in \{0,1\}^{2^\sigma}$ which is supposed to equal $C_i(a) = \langle a, y_i \rangle$ for every $a \in \{0,1\}^\sigma$.
2. For every constraint $\psi(y_i, y_j)$ consider the corresponding quadratic system $Q(y_i, y_j, z_{ij})$ where z_{ij} takes values in $\{0,1\}^{O(2^{2\sigma})}$. Provide an encoding C_{ij} for z_{ij} where $C_{ij}(a)$ is supposed to equal $\langle a, z_{ij} \rangle$, as well as an encoding D_{ij} of Q, where for every linear combination b of quadratic terms in the variables y_i, y_j, z_{ij}, $D_{ij}(b)$ is supposed to equal the value of this combination.

The verifier of Ψ' chooses a random constraint $\psi(y_i, y_j)$ in Ψ and runs the PCP from last lecture on the part of the proof that contains the encodings C_i, C_j, C_{ij}, D_{ij} to verify that the constraint is satisfied.[1]

Clearly if Ψ is satisfiable, the verifier of Ψ' will accept with probability 1. Now we argue that if Ψ' rejects with probability at most $\delta/2$, then some sassignment violates at most a δ-fraction of constraints in Ψ.

Assume Ψ' rejects with probability at most $\delta/2$. Let y_i be the most likely assignment encoded by C_i (i.e. the one such that the encoding of y_i and C_i differ in the smallest number of places, breaking ties arbitrarily). Then for at least a $1-\delta$ fraction of the constraints ψ, when ψ is chosen Ψ' accepts with probability at least $1/2$. By the analysis from last time if this is the case, then all of C_i, C_j and C_{ij} must be $1/8$-close to encodings of some assignments y_i', y_j' and z_{ij} so that $Q(y_i', y_j', z_{ij})$ is satisfied and therefore $\psi(y_i', y_j') = 1$. Since y_i is the most likely assignment encoded by C_i, it must be that the encodings of y_i and y_i' differ in at most a $1/4$-fraction of places. But any two distinct linear functions differ on at least half the outputs, so it must be that $y_i = y_i'$. Similarly $y_j = y_j'$. Therefore y satisfies the constraint ψ, so it satisfies a $1-\delta$ fraction of constraints of Ψ.

[1] This is not exactly the same PCP. In the last lecture C_i, C_j, and C_{ij} were grouped into a single chunk, while here they are separate. However we can run the linearity test and local decoding procedures on each part separately with the same effect.