Connectivity in Logspace.

We saw v prob. vector then $\| A^k v - \mathbb{1}/n \| \leq \lambda^k$.

This gives an algorithm for checking $\overset{\text{undir.}}{}$connectivity in expanders:
(ie. given s,t, $\exists$ path $s \leftrightarrow t$?).

lemma: if s,t connected then exists path $s \leftrightarrow t$ of len $< \ell = O(\log n)$.

proof: $\| A^k v - \mathbb{1}/n \|_1 \leq \sqrt{n} \cdot \| A^k v - \mathbb{1}/n \| \leq \sqrt{n} \cdot \lambda^k . \leq 1/2n$

then no component can be $< 1/n - 1/2n = 1/2n$, so enough to
check all random walks of len $\ell . = \log(1/2n\sqrt{n}) / \log \lambda$.

If graph not expander, can use randomized algorithm instead:
do random walk for $100n^4$ steps from s.
if t seen: answer yes.
o/w answer no.

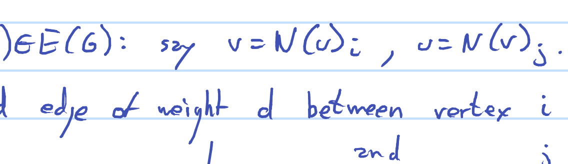
Can analyse using that G connected $\Rightarrow \lambda < 1 - \frac{1}{12n^2}$. Instead
will see how to make algorithm deterministic.
Plan: transform every c.c. of G into an expander.

Could do G^K, but that increases degree too much...

Construction: replacement product.
G $(n, D, 1-\varepsilon)$-expander H $(D, d, 1-\delta)$-expander

$G \circledR H$ $(n \cdot D, 2d, 1 - \varepsilon\delta^2/24)$-expander.

e.g.



construction: for every $v \in G$, have a copy of H.

if $(u,v) \in E(G)$: say $v = N(u)_i$, $u = N(v)_j$.
add edge of weight d between vertex i of H_u
and j H_v.
$\downarrow$
d parallel edges.

lemma:

Let us use construction to prove:

th (Reingold) UPATH $\in L$.

1. reduce degree of G to 4, by replacing every vertex with a cycle.
2. increase degree of G (add self-loops) so that a
$(d^{50}, d/2, 1/100)$-expander exists. Call it H.
$\hookrightarrow$ extremely good!
3. let $G_0 = G$, $G_k = (G_{k-1} \circledR H)^{50}$.

obs well-defined: $\deg(G_{k-1} \circledR H) = d$, $\deg(\sim^{50}) = d^{50}$.

claim: every cc of $G_{10 \log n}$ is an expander
more precisely $G_k \rightarrow (d^{50k} \cdot n, d^{50}, 1-\varepsilon)$-exp,
$\varepsilon = \min(1/20, (1.5)^k/(12n^2))$.

proof. $\lambda(G_{k-1}) \leq 1-\varepsilon \rightarrow \lambda(G_{k-1} \circledR H) \leq 1 - \frac{\varepsilon \cdot (1-1/100)^2}{24} \leq 1 - \varepsilon/25$

$\lambda(G_k) \leq (1-\varepsilon/25)^{50} \leq (1-2\varepsilon)$.
$\nwarrow$ $\varepsilon < 1/20$. //

so algorithm is to run s-t-conn over $G_{10 \log n}$.
but... cannot store new graph in logspace!
no problem, all we need is, given id of $v \in V(G_{10 \log n})$
and id of edge, $i \in [d^{50}]$, $[d^{500 \log n} \cdot n]$
find id of i-th neighbour of v in logspace.

obs if we can do one step, then we can do ℓ steps.

obs $G_k = (G_{n-1} \circledR H)^{50}$, so enough to slow how to do
a stop in $G \circledR H$.

we have (u,v) and $i \in [d]$.
if $i \leq d/2$, we more within H: $(u,v) \mapsto (u, N_H(v)_i)$.
if $i > d/2$, we more within G, through one of $d/2$ edges
(does not matter which).
$(u,v) \mapsto (u',v')$ st $u' = N_G(u)_v$ and $N_G(u')_{v'} = v$.

do induction etc all is well. ✱

proof of replacement lemma

claim: $A \circledR B = 1/2 \hat{A} + 1/2 (I_n \otimes B)$

$\hat{A}$ has a 1 in $(u,v), (u',v')$ iff   and 0 o/w

we'll slow $\lambda((G \circledR H)^3) \leq 1 - \frac{\varepsilon\delta^2}{8} . \leq (1 - \frac{\varepsilon\delta^2}{24})^3$.

claim: $B = (1-\delta)B' + \delta J_D$ $\|B'\| \leq 1$, $J_D = D \times D$ all 1s$/D$.

expand $C = (A \circledR B)^3$, got $C = (1 - \frac{\delta^2}{8}) \underset{\|C'\|=1}{\underset{\llcorner}{C'}} + \frac{\delta^2}{8} \underbrace{(J_n \otimes J_D) \hat{A} (I_n \otimes J_D)}_{A \otimes J_D \text{ (claim)}}$.

$\uparrow$
$\lambda = 1-\varepsilon$.

$\lambda(C) \leq (1 - \frac{\delta^2}{8}) \cdot 1 + \frac{\delta^2}{8}(1-\varepsilon)$
$= 1 - \frac{\varepsilon\delta^2}{8}$ //.