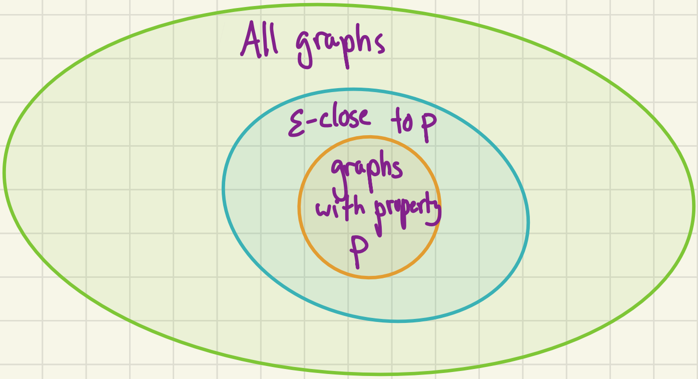
Lecture 7+8

- Property Testing
- Testing Planarity
- Partition oracles

Property Testing



P is a subset of graphs

Can we distinguish graphs in P from graphs that are not in P? not even ε-close?

$\Big\{$ in sublinear time?

Goal: if G has property P, pass
if G ε-far from P, fail

(if G is ε-close, can either Pass or fail)

For today:

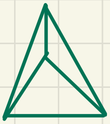
assume Δ is constant

def deg ≤ Δ graph G is ε-close to P if can remove some ≤ $\varepsilon \Delta n$ edges to turn G into $G' \in P$

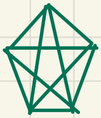
Planarity:

def a Planar graph can be drawn in plane
st. edges intersect only at endpts.

e.g.



planar



not planar

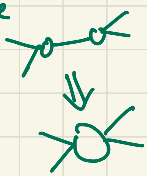
Cool Thm [Kuratowski]

G is planar iff does not contain K_5 or $K_{3,3}$ as minor

↑ Complete graph on 5 nodes

↑ Complete bipartite graph with 3 nodes on side

↑ subgraph repeatedly contract edges into nodes



How to test planarity?

↗ in sublinear time

↑ $P = \{ G \mid G \text{ is planar} \}$

Hyperfiniteness

def G is (ε, k)-hyperfinite if can remove $\le \varepsilon n$ edges + remain with all connected components of size $\le k$. ← k can be fctn of ε

remove few edges + break up graph into tiny pieces.

Important Thm $\forall \varepsilon, \Delta$, $\exists$ constant c s.t. every planar graph of max degree Δ is $(\varepsilon\Delta, c/\varepsilon^2)$-hyperfinite

similar theorem for any class of graphs defined by "forbidden minors"

Note subgraphs of deg $\le \Delta$ planar graphs are also deg $\le \Delta$ planar graphs
$\Rightarrow$ also hyperfinite

Why is hyperfiniteness useful?

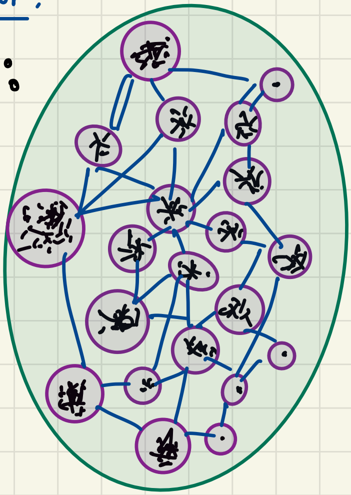
how in sublinear time?

↓

Partition G with parameters $\left(\frac{\varepsilon}{2} \cdot \Delta, \frac{c}{\varepsilon^2}\right)$

$G' \leftarrow G$ minus edges between partitions

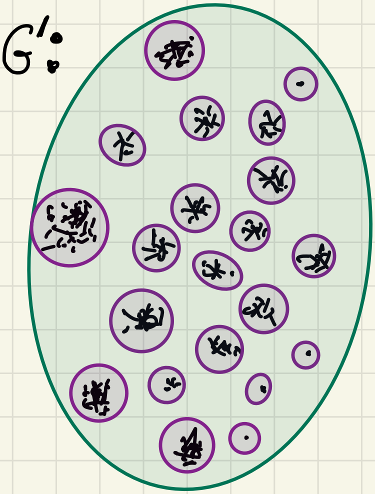
G:



nice properties of G':

- G planar $\Rightarrow$ G' planar

G':



- G' is very similar to G:

  differ by $\leq \frac{\varepsilon}{2}\Delta$ edges

  $\Rightarrow$ if G is ε-far from planar then G' is $\frac{\varepsilon}{2}$-far from planar

- All connected components in G' are size $O(1/\varepsilon^2)$

↖ Can test for planarity in time indep of n

If there is no such partition of G, then G is not planar !! but, can we find it in sublinear time?

Partition Oracle

(use slightly different parameters from previous)

input v ← node

output $P[v]$ ← name of v's partition

$$\text{s.t. } \forall v \in V \qquad \left.\begin{array}{l} (1)\ \ |P[v]| \le k \\ (2)\ \ P[v] \text{ connected} \end{array}\right\} \text{partitions small \& connected}$$

& if G planar then

$$(\text{with prob} \ge \tfrac{9}{10}) \quad \underbrace{|\{(u,v)\in E \mid P(u) \ne P(v)\}|}_{\text{edges crossing partitions}} \le \underbrace{\frac{\varepsilon \Delta n}{4}}_{\text{small fraction}}$$

Note for planar graphs there is at least one P, but there could be many possible partitions P. the oracle doesn't have to decide "in advance" which partition to use, but must answer consistently (without memory of past or knowledge of future queries)

a main difficulty →

Algorithm given partition oracle P

↑ assume it always works for planar G

I. Does P give partition that "looks right"? (e.g. few crossing edges)

need to sample random edges, easy since degree Δ is constant

- $\hat{f} \leftarrow$ estimate of # edges (u,v) s.t. $P[u] \neq P[v]$ to within additive error $\leq \frac{\varepsilon \Delta n}{8}$ with prob of failure $(\delta) \leq \frac{1}{10}$
- if $\hat{f} \geq \frac{3}{8} \varepsilon \Delta n$, output "not planar" & halt

II. Test random partitions for planarity

- Choose $S = O(\frac{1}{\varepsilon})$ random nodes
  - $\forall s \in S$ if $P[s]$ not planar output "not planar" & halt

  } size $\leq k$ so easy to test
- Output "planar"

Runtime: part I: $O(\frac{\Delta}{\varepsilon^2})$ calls to oracle

part II: $O(\Delta/\varepsilon^2)$ calls to determine $P[s]$ via BFS

$O(\Delta/\varepsilon^3)$ total calls

Analysis: (assume oracle works for planar G)

<u>def</u> $C = \{(u,v) \in E \mid P(u) \neq P(v)\}$ "edges that cross between partitions"

I. if G is planar, $\exists$ good partition & by assumption that oracle works, # edges crossing partition $\le \frac{\varepsilon \Delta n}{4}$

so $E[\hat{f}] \le \frac{\varepsilon \Delta n}{4}$

$\Rightarrow$ (by Chernoff/Hoeffding) $\hat{f} \le \frac{\varepsilon \Delta n}{4} + \frac{\varepsilon \Delta n}{8} = \frac{3\varepsilon \Delta n}{8}$

with prob $\ge 9/10$

$\Rightarrow$ algorithm continues to part II with prob $\ge 9/10$

Also, $\forall s \in V$, $P[s]$ is Planar

$\Longrightarrow$ output "Planar" with prob $\ge 9/10$

II. If G is ε- far from planar:

Case 1 Partition P doesn't satisfy $|C| < \frac{\varepsilon \Delta n}{2}$:

sampling bounds $\Rightarrow$ $\hat{f} > \frac{\varepsilon \Delta n}{2} - \frac{\varepsilon \Delta n}{8} = \frac{3}{8} \varepsilon \Delta n$

$\Rightarrow$ output "not planar" with prob $\geq 9/10$

Case 2 P satisfies $|C| < \frac{\varepsilon \Delta n}{2}$:

$G' \leftarrow G$ with edges in C removed

note: G' is $\frac{\varepsilon}{2}$-close to G

so G is ε-far from planar,

$\Rightarrow G'$ is $\frac{\varepsilon}{2}$-far from planar

issue: we are picking random nodes in part II, not random edges.

but graph is degree $\leq \Delta$

since G' $\frac{\varepsilon}{2}$-far from planar, must change $\geq \frac{\varepsilon}{2} \cdot \Delta n$ edges, which touch $\geq \frac{\varepsilon}{2} \cdot n$ nodes so with prob $\geq \frac{\varepsilon}{2} \cdot n$, pick node in component which is not H-minor free.

▨

Remaining issue:

How do we implement $\mathcal{P}$?

Plan:

1) Define <u>global</u> partitioning strategy
2) Figure out how to <u>locally</u> implement (only find partition of given node, not whole solution).

Useful concept: Isolated Neighborhoods

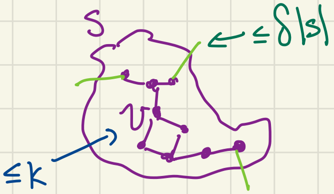
def S is (δ, k)-isolated neighborhood of node v if:

(1) $v \in S$

(2) S connected

(3) $|S| \leq k$

(4) # edges connecting S + $\bar{S} \leq \delta |S|$



↑ $\delta < 1$ but degree bound only gives $\Delta \cdot |S|$

Observe in planar graphs, in any good partition, most nodes have (δ, k)-isolated nbhds

$\varepsilon \cdot \Delta$ → δ k ← c/ε^2

obvious? yes, on average but

Planar G is hyperfinite $\Rightarrow \exists$ partition with few total crossing edges

but maybe some parts have lots of edges coming out? still, most have close to average. (Markov's ≠) * think at home

Will need observation to be true in context of evolving "step-by-step" partition.

Luckily, graph stays planar | hyperfinite as evolve (remove nodes)

Global Partitioning Algorithm ← mental thought process

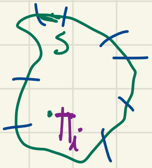
(note: will use smaller δ & k will be slightly bigger, but only differ by constants)

```
Let $\pi_1 \dots \pi_n$ be nodes in random order
$P \leftarrow \phi$
For $i = 1..n$ do
    if $\pi_i$ still in graph then
        if $\exists\, (\delta, k)$-isolated nbhd of $\pi_i$
              in remaining graph
        then $S \leftarrow$ this nbhd
        else $S \leftarrow \{\pi_i\}$
    $P \leftarrow P \cup \{S\}$
    Remove $S$ & adjacent edges from $G$
```

BFS to distance k of π_i

can't find S with few connecting edges →

⇐ S is just one node. hopefully doesn't happen often!



↑ crossing edges

Behavior: few crossing edges?

- S s.t. S is (δ, k)-isolated contribute $\le \delta|S|$ edges
  $\Rightarrow$ overall $\sum \delta |S_i| \le \delta \cdot n$
- S s.t. $S = \{\pi_i\}$ (one node) contributes $\Delta \Rightarrow$ overall Δn? ☹
  need to show not too many!!
  if $\le \varepsilon' n$ such nodes, contribute $\le (\varepsilon' n)\Delta$ total

no bad way to "rip out" isolated nbhds

$|V|=n$ $\downarrow$

<u>Lemma</u> if $G'=(V',E')$ subgraph of planar $G=(V,E)$ s.t. $|V'| \geq \delta n$

then $\leq \frac{\varepsilon}{30}$ fraction of V' don't have (δ,k)-isolated nbhds for $\delta = \varepsilon\Delta/30$

$k = \Theta(1/(\varepsilon\Delta)^2)$ } need to use stronger settings in "important theorem"

<u>Pf idea</u>

G planar

$\Downarrow$

G' planar

$\Downarrow$

G' hyperfinite

$\Downarrow$

$\exists$ partition s.t. <u>most</u> nodes in (k,δ)-isolated nbhd — Markov's $\neq$

$+$

π_i randomly chosen

$\Downarrow$

whp π_i in (k,δ)-isolated nbhd

∎

So, not <u>too</u> many "singletons"

Local Simulation of Partition Oracle

- input v
- assume access to random fctn $\pi(v)$ st. $\pi: V \to [n]$
- Output $P[v]$

Algorithm on input v

I. recursively compute $P[w]$ $\forall$ w st.

why does recursion stop?

note: only recurse on lower ranked nodes

$\pi(w) < \pi(v)$

$\&$ dist w from $v \leq 2k$

$\Delta^{O(k)}$ of these

II. if $\exists$ w st $v \in P[w]$ ← $P[v]$ already decided by earlier w

(A) then $P[v] = P[w]$

(B) else look for (k, δ)-isolated nbhd of v ← can't be in nbhd of any node outside of 2k radius ball

(ignoring nodes in $P[w]$ for smaller w's)

if find one, $P[v] \leftarrow$ this nbhd

else $P[v] \leftarrow \{v\}$

Implementing algorithm:

step I:

$\Delta^{O(k)}$ recursive computations on lower ranked nodes. Analysis similar to "greedy" lecture $\Rightarrow 2^{\Delta^{O(k)}} + k \approx O(1/\varepsilon^3)$

step II: (A) computed in step I

(B) figure out remaining nodes w/in dist K from step I.

Brute force on graph of size $\leq \Delta^{O(k)}$

Can do much better:

poly(Δ/ε) possible!