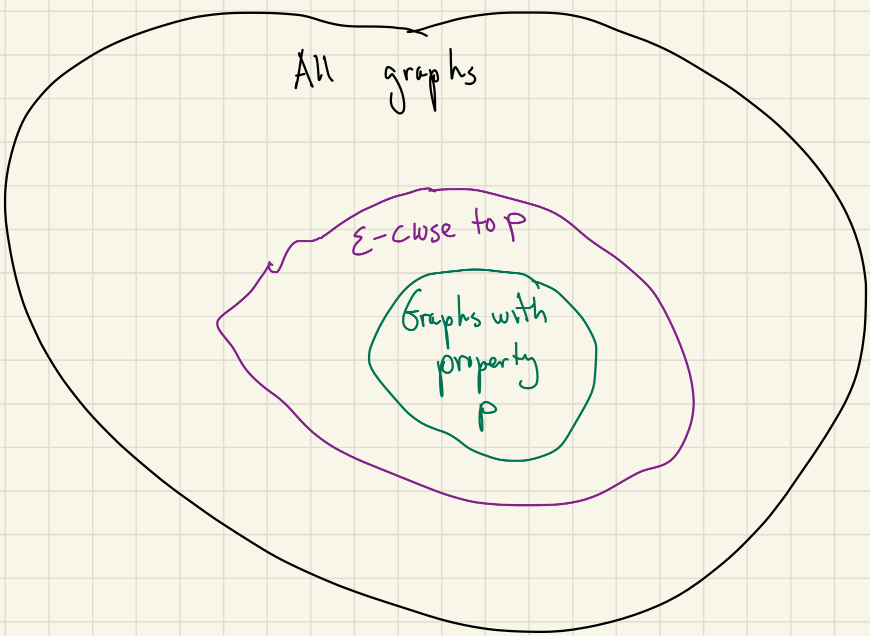
Lecture 6:

Property testing:

is the graph planar?

Property Testing

examples of P:
- planar
- bipartite
- no small cuts
- no triangles
- Connected



Can we distinguish graphs with property P from <u>far</u> from P?

e.g. G is ε-far from planar if must remove $\geq \varepsilon \cdot \Delta \cdot n$ edges to make it planar

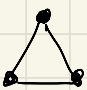
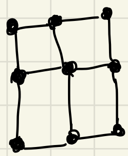
Δ is max degree

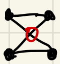
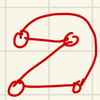
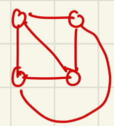
Today's goal:

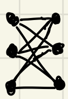
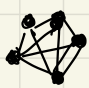
test planarity in time independent of n
(but exponential in ε)

for graphs with max degree Δ

what is a planar graph?
Can be drawn on plane s.t. edges don't intersect

K_3   Yes!

$K_{2,2}$  K_4    Yes!

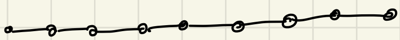
$K_{3,3}$  K_5  NO

Thm [Kuratowski]

G is planar iff G is $K_{3,3}$ + K_5 minor free

Testing Planarity:

Can't hope to distinguish

 from 

in sublinear time!

def G is ε-close to planar iff

can remove $\leq \varepsilon \cdot \Delta n$ edges to make it

$\begin{cases} \text{planar} \\ K_{3,3} + K_5\text{-free} \end{cases}$ equivalent

else G is ε-far

Goal: Given G

- if G planar, PASS
- if G ε-far from planar, FAIL

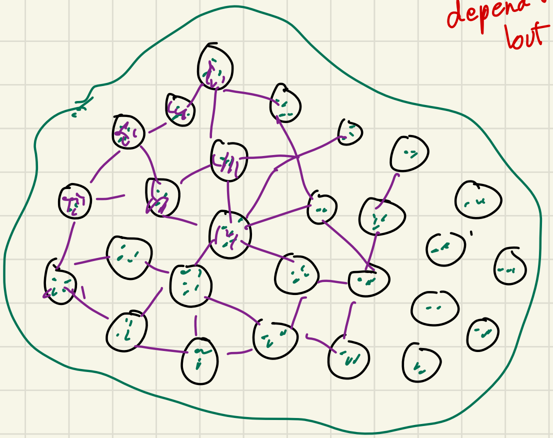
Plan for tester: use nice property of Planar graphs

Can always remove $\underbrace{\text{small fraction}}_{\leq \varepsilon}$ of edges

& break up graph into tiny connected components

const size
depend on ε
but not n

G:



def. G is "(ε, k)-hyperfinite" if

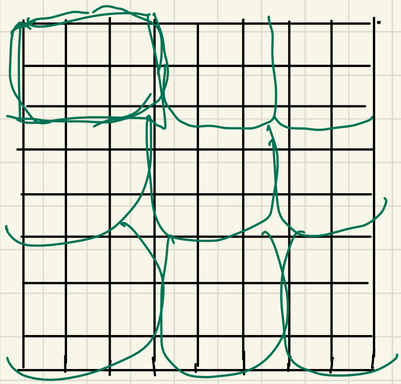
Can remove $\leq \varepsilon \cdot n$ edges &

remain with connected components

of size $\leq k$ nodes

def. G is "(ε, k)-hyperfinite" if
Can remove $\leq \varepsilon n$ edges &
remain with connected components of
size $\leq k$

Example: $\underbrace{m \times m}_{n = m^2}$ grid graph



break into $l \times l$ squares:

size of component $\approx l^2 \approx \frac{16}{\varepsilon^2}$ ← does not depend on n

# of components $\approx \left(\frac{m}{l}\right)^2$

edges crossing border of one component $\approx 4l$

total # edges crossing borders $\approx (4l) \cdot \left(\frac{m}{l}\right)^2$
$\approx \frac{4 \cdot m^2}{l} \approx \varepsilon m^2$ ← $\leq \Theta(\varepsilon)$ fraction of edges

$l = \frac{4}{\varepsilon}$

not just grid graphs!

def. G is "(ε, k)-hyperfinite" if can remove $\leq \varepsilon n$ edges & remain with connected components of size $\leq k$

Useful Thm

$\forall\ 0<\varepsilon<1$, every planar graph G of $\deg \leq \Delta$ is $\left(\varepsilon \cdot \Delta, \frac{c}{\varepsilon^2}\right)$-hyperfinite

note subgraphs of planar graphs are also planar, so also hyperfinite but only remove # edges in proportion to # nodes in subgraph

$\Rightarrow$ can recurse & break up further

Why does hyperfiniteness help in testing?

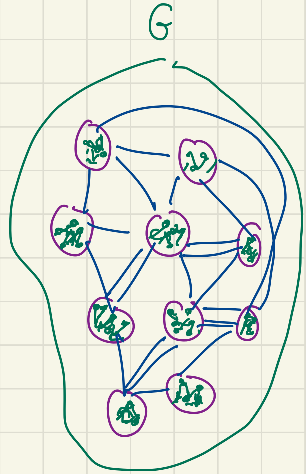
Plan for testing paradigm:

1) Partition graph G into G'

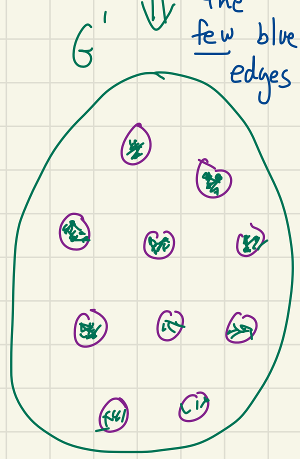
how in sublinear time?

- Only const size com. comp. remain
- removed few edges $(\leq \varepsilon \cdot \Delta n)$

if can't do this, then G is not planar!



remove the few blue edges



2) If G' is close to having property so is G

const time

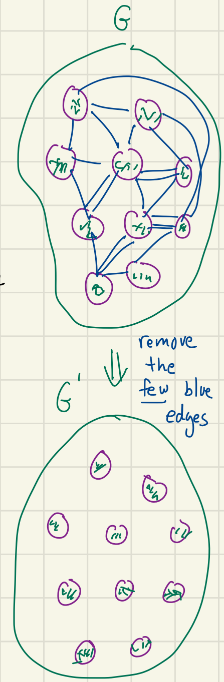
- so test G' by picking random component & seeing if it is planar

Easy to test since collection of const sized graphs!!

1) Partition graph G into G'

how in sublinear time?

- only const size com. comp. remain
- removed few edges $(\le \varepsilon \cdot \Delta \cdot n)$
- if can't do this, G is <u>not</u> H-minor free



2) If G' is close to having property so is G

const time

- so test G' by picking random components + seeing if they have the property

Easy to test since collection of const sized graphs!!

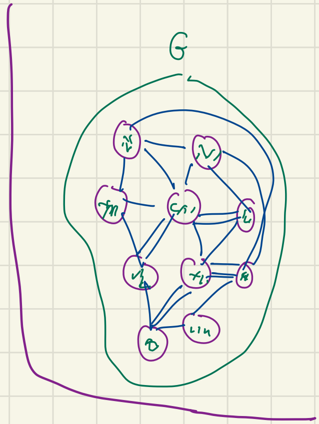
How to determine G'?

- need "local" (sublinear) way to figure out if edge is "blue" (between components) or "green" (within components)

- will do even better!

give oracle that tells you for each node the "name" of its component

Partition Oracle:



input: node v

output: name of v's partition $P[v]$

s.t. $\forall v \in V$ (1) $|P[v]| \leq K$ (small)

(2) $P[v]$ is connected component

+ if G planar then with prob $\geq 9/10$

(3) $|\{(u,v) \in E \mid P(u) \neq P(v)\}| \leq \frac{\varepsilon \cdot \Delta \cdot n}{4}$ (few crossing edges)

Algorithm given Partition Oracle:

I. Does partition oracle give partition that "looks right"?

- let $\hat{f} \leftarrow$ estimate of # edges (u,v) s.t. $P[u] \neq P[v]$ to additive error $\leq \frac{\varepsilon \cdot \Delta \cdot n}{8}$ (with prob of failure $\leq 1/10$)
- if $\hat{f} > \frac{3}{8}\varepsilon \cdot \Delta \cdot n$, output FAIL + halt

II. Test random partitions

- choose $S = O(1/\varepsilon)$ random nodes

  if for any $s \in S$, $\underbrace{P[s] \text{ not planar}}_{\text{const size so easy}}$

  reject + halt

III. Accept anything that passed up to this point

Runtime (given oracle):

Part I: $O\left(\frac{1}{\varepsilon^2}\right)$ calls to oracle

Part II: $O\left(\underbrace{\frac{1}{\varepsilon}}_{|S|} \cdot \underbrace{\Delta \cdot k}_{\text{calls to oracle for BFS}}\right)$ " " " $= O\left(\frac{\Delta}{\varepsilon^3}\right)$

$k \approx O\left(\frac{1}{\varepsilon^2}\right)$

Algorithm given Partition Oracle:

I. Does partition oracle give partition that "looks right"?
e.g. few crossing edges

- $\hat{f} \leftarrow$ estimate of # edges (u,v) s.t. $P[u] \neq P[v]$
  to additive error $\leq \frac{\varepsilon \Delta n}{8}$ (with prob of failure $\leq \frac{1}{10}$)
- if $\hat{f} > \frac{3}{8} \varepsilon \Delta n$, output "FAIL" & halt

II. Test random partitions

these choose random partitions

- Choose $S = O(1/\varepsilon)$ random nodes
  - if for any $s \in S$, $P[s] \geq k$ or $\underbrace{P[s] \text{ not planar}}_{}$
    reject & halt

  constant size $k = O(1/\varepsilon^2)$ so easy to test

III. Accept anything that passed up to this point

Behavior (assuming P always "correct"):

- if G is planar:
  1) $E[\hat{f}] \leq \frac{\varepsilon \Delta n}{4}$

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

  $\Rightarrow$ algorithm should pass stage I (with prob 9/10)

  2) $\forall s \in V$ $P[s]$ is planar

  $\Rightarrow$ algorithm never fails stage II

$\Rightarrow$ PASS

Algorithm given Partition Oracle:

I. Does partition oracle give partition that "looks right"?
e.g. few crossing edges

- $\hat{f} \leftarrow$ estimate of # edges (u,v) s.t. $P[u] \neq P[v]$ to additive error $\leq \frac{\varepsilon \Delta n}{8}$ (with prob of failure $\leq \frac{1}{10}$)
- if $\hat{f} > \frac{3}{8}\varepsilon\Delta n$, output "FAIL" & halt

II. Test random partitions

these choose random partitions

- Choose $S = O(1/\varepsilon)$ random nodes
  - if for any $s \in S$, $|P[s]| \geq k$ or $\underbrace{P[s] \text{ not planar}}_{\text{constant size}}$ reject & halt

$k = O(1/\varepsilon^2)$ so easy to test

III. Accept anything that passed up to this point

Behavior (assuming P always "correct"):

- if G is ε-far from planar: let $C = \{(u,v) \in E \mid P[u] \neq P[v]\}$

<u>Case 1</u> $|C| > \frac{\varepsilon \Delta n}{2}$

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

$\Rightarrow$ output "fail" with prob $\geq 9/10$

<u>Case 2</u> $|C| < \frac{\varepsilon \Delta n}{2}$

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

since G is ε-far from planar &

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

Then G' must ~~be~~ be $\frac{\varepsilon}{2}$-far from planar

must remove $\geq \frac{\varepsilon \Delta n}{2}$ edges to make planar

which touch $\geq \frac{\varepsilon n}{2}$ nodes

so with prob $\geq \frac{\varepsilon}{2}$ pick a

node in $\underbrace{\text{component that needs to be modified}}_{\text{not planar!}}$

algorithm will detect that component not planar!

$\blacksquare$

But how do we implement P?

Plan for designing Partition Oracle:

1) Define <u>Global</u> partitioning strategy (not sublinear time)

2) Figure out how to implement it <u>locally</u>
   (only find partition of a given node, not whole solution)

Useful Concept: "Isolated" neighborhoods

<u>def.</u> G is "(ε, k)-hyperfinite" if can remove $\leq \varepsilon n$ edges & remain with connected components of size $\leq k$

<u>def</u> S is (δ, k)-isolated nbrhd of node v

if
1) $v \in S$
2) S connected
3) $|S| \leq k$
4) # edges connecting S & $\bar{S}$ is $\leq \delta \cdot |S|$

<u>Note</u>: In hyperfinite graphs, most nodes have (δ, k)-isolated nbrhds