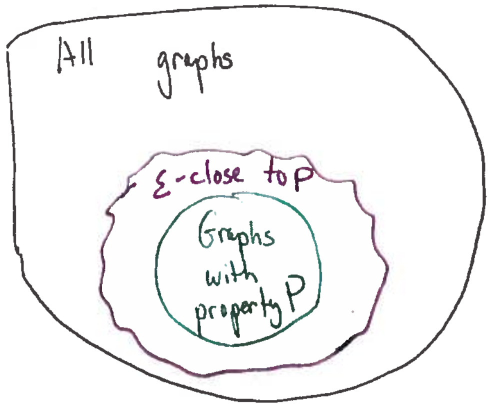
Lecture 6:

- Testing Planarity + Minor-freeness
- Partition Oracles

Property Testing



Can we distinguish? in sublinear time?

e.g. P = "planar"

Compromise

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

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

Today: Test planarity in time independent of n (but exponential in ε)

Testing H-minor freeness

all graphs have max degree $\le d$

def.
- H is "minor" of G
  if can obtain H from G via
  vertex removals, edge removals, edge contractions

- G is "H-minor-free" if H not minor of G

- G is "ε-close to H-minor-free" if
  can remove $\le \varepsilon dn$ edges to make it
  H-minor-free
  (o.w. G is "ε-far")
- minor closed property P –
  if $G \in P$ then all minors of G are in P

Really Cool Theorem [Robertson + Seymour]
Every minor-closed property is expressible
as a constant # of excluded minors.

Some minor-closed properties: $K_{3,3}$ or K_5
NO
planar graph, bounded tree width, ...

Goal: Testing H-minor freeness
Pass H-minor free graphs
Fail if far from H-minor free

more definitions

• G is "(ε, k)-hyperfinite" if

k can be a function of ε

Can remove $\leq \varepsilon n$ edges

+ remain with connected components of size $\leq k$

(ie., can remove few edges and break up graph into very small components.)

Useful Thm

Given H

$\exists\, C_H$ st. $\forall\, 0 < \varepsilon < 1$, every H-minor free graph of deg $\leq d$

constant that depends only on H

is $(\varepsilon d, C_H^2/\varepsilon^2)$- hyperfinite.

(ie. remove $\underbrace{\leq \varepsilon d n}_{\leq \varepsilon \text{ fraction}}$ edges + components of size $\underbrace{O}_{\text{independent of } n}(1/\varepsilon^2)$)

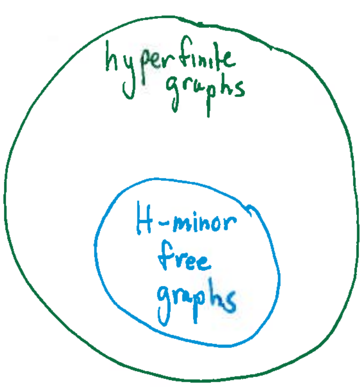
note

Subgraphs of H-minor free graphs also H-minor free

+ so also hyperfinite

but, only remove #edges in proportion to #nodes in subgraph

$\Rightarrow$ Can "recurse" + break up further



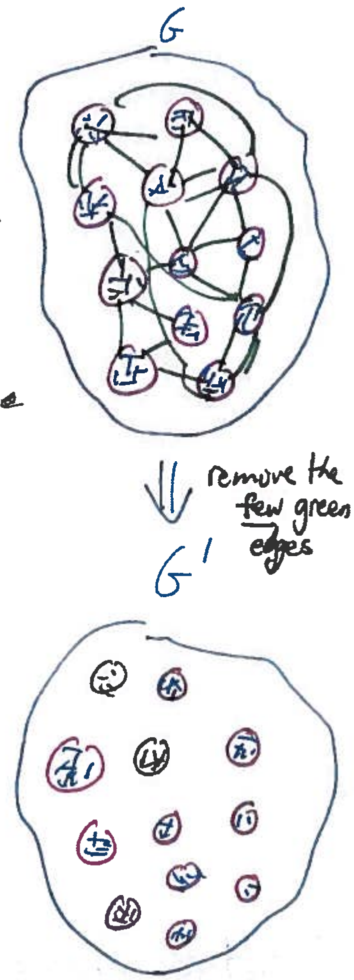
Why is hyperfiniteness useful?

Partition graph G into G'

how in sublinear time? {
- only const size connected components remain
- removed only few edges ($\le \varepsilon d n$)
- if can't do this, G is not H-minor free
}

If G' is close to having property, so is G

Constant time { so test G' by picking random components & seeing if they have the property }



Easy to test since collection of constant sized graphs!!

Need a "local" (sublinear) way to determine G'. For now assume we have "partition oracle" P

(with parameters $\frac{\varepsilon d}{4}$, k)

fraction edges removed ↗ $\frac{\varepsilon d}{4}$; component size ↖ k

<u>input</u>: vertex v
<u>output</u>: $P[v]$ (v's partition name)

s.t. $\forall v \in V$ (1) $|P[v]| \le k$
(2) $P(v)$ connected

} partitions small + connected

& if G is H-minor free

(with prob $\ge \frac{9}{10}$) $|\{(u,v) \in E \mid P(u) \ne P(v)\}| \le \frac{\varepsilon d n}{4}$

} remove only few edges

Algorithm given partition oracle P:

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

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

II. Test random partitions

- Choose $S = O(1/\varepsilon)$ random nodes ← these select "random" partitions
  - if for any $s \in S$, $P[s] \geq k$ or $P[s]$ not H-minor free, reject + halt

    size $k \leq O(\frac{1}{\varepsilon^2})$
    so easy to test
- Accept

Runtime:

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

Part II: $O(\frac{d}{\varepsilon^2})$ calls to oracle to determine $P[s]$

$O(\frac{d}{\varepsilon^3})$ total calls

Analysis (assume oracle P always correct)

- if G is H-minor free:

  1) $E[\hat{f}] \lessapprox \frac{\varepsilon d n}{4}$

     sampling bounds (Chernoff/Hoeffding) $\Rightarrow \hat{f} \lessapprox \frac{\varepsilon d n}{4} + \frac{\varepsilon d n}{8} = \frac{3}{8}\varepsilon d n \Rightarrow$ algorithm doesn't fail at stage I with prob $\geq \frac{9}{10}$

  2) $\forall\ S \subseteq V,\ P[S]$ is H-minor free

- if G is ε-far from H-minor free:

  Case 1: P's output doesn't satisfy $|\{(u,v)\in E : P(u) \neq P(v)\}| \lessapprox \frac{\varepsilon d n}{2}$

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

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

  Case 2: P satisfies $C \equiv |\{(u,v)\in E : P(u) \neq P(v)\}| < \frac{\varepsilon d n}{2}$

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

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

  so, if G is ε-far from having property then G' is $\frac{\varepsilon}{2}$-far from having property!

Since G' is $\frac{\varepsilon}{2}$ – far from H-minor free

must change $\geq \frac{\varepsilon d n}{2}$ edges, which touch $\geq \frac{\varepsilon n}{2}$ nodes

So, with prob $\geq \frac{\varepsilon}{2}$, pick node in component which is not H-minor free ∎

Remaining Issue:

Implementing partition oracle P

Plan:

1) Define Global partitioning strategy (not sublinear time)

2) Figure out how to implement locally (only find partition of given node, not whole solution)

A 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 & $\bar{S}$ $\leq \delta |S|$

In hyperfinite graphs, most nodes have (δ, k)-isolated nbhds.

Is this obvious?

- G hyperfinite $\Rightarrow \exists$ partitioning
- but will need this to be true about remaining graph in context of algorithm that may find a different partition "step-by-step"

– luckily, no matter what was removed earlier, we still have an H-minor free graph so still hyperfinite!

Global Partitioning Algorithm ⟸ a "mental thought process"

Let $\pi_1 \cdots \pi_n$ be nodes in random order

$\mathbb{P} \leftarrow \phi$

For $i = 1..n$ do

if π_i still in graph then

use $\delta = \frac{\varepsilon d}{4}$, $k = \theta\left(\frac{1}{\varepsilon^2}\right)$

how? need to consider all nodes within distance k of π_i

if $\exists$ (δ, k)-isolated nbhd of π_i in remaining graph

then $S \leftarrow$ this nbhd

else $S \leftarrow \{\pi_i\}$

S is just one node in this case. hopefully doesn't happen often!

$\mathbb{P} \leftarrow \mathbb{P} \cup \{S\}$

Remove S + adjacent edges from graph

Does this give a partition with few crossing edges?

- S s.t. S is (δ, k)-isolated contribute $\le \delta |S|$ edges

  which overall $\le \delta \cdot n$

- S s.t. $S = \{\pi_i\}$ (one node):

  need to show that not too many of these!

<u>Lemma</u> if G' is subgraph of a (hyperfinite) graph G s.t. G' has $\geq \delta n$ nodes then $\leq \frac{\varepsilon}{30}$ fraction of nodes in G' don't have (δ, k)- isolated nbhds, for $\delta = \varepsilon/30$
$k = \Theta(\varepsilon^3)$

<u>Pf idea</u>

G H-minor free

$\Downarrow$

G' H- minor free

$\Downarrow$

G' hyperfinite

$\Downarrow$

$\exists$ partition s.t. most nodes in G' are in (k, δ)- isolated nbhd

$+$

π_i randomly chosen in G'

$\Downarrow$

whp π_i in (k, δ)-isolated nbhd. ∎

So, not too many "singletons" !!

Local Simulation of Partitioning Oracle:

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

$d^{O(k)}$ of these

- recursively compute $P[w]$ for all w s.t.
  - $\pi(w) < \pi(v)$
  - w is distance $\leq 2k$ from v
- if $\exists$ w s.t. $v \in P[w]$
  then $P[v] = P[w]$

  else look for (k,δ)-isolated nbhd of v

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

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

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

Query Complexity: $2^{d^{O(k)}}$ using analysis from last time $+ k \times \Theta(\varepsilon^3)$

but can do much better:

currently $d^{O(\log^2(1/\varepsilon))}$ is possible