Lecture 8:

Testing dense graphs

- bipartiteness

Adjacency Matrix model

G represented by matrix A
s.t. can query A in
one step

$$A = \begin{bmatrix} & A_{ij} & \end{bmatrix}$$

1 if $(i,j) \in E$
0 o.w.

Distance from property P:

<u>def</u> G is ε-far from P if must change $> \varepsilon \cdot n^2$
entries in A to turn G into member of P

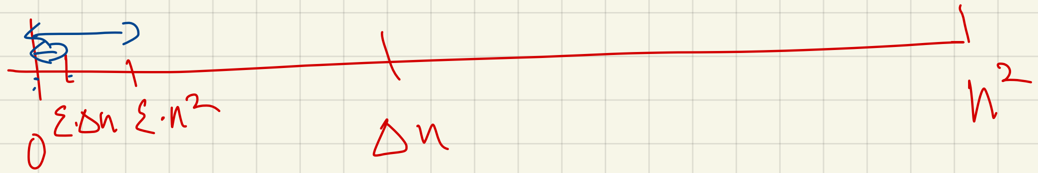
Testing "sparse" properties:

<u>all</u> graphs are ε-close to connected in this model

$\Rightarrow$ trivial tester outputs "PASS" w/o looking at graph

	Graph type	max degree	natural representation	notion of distance
Previously	sparse	Δ	adjacency list	$\le \varepsilon \cdot \Delta \cdot n$ edges changed
Now	dense	n	adjacency matrix	$\le \varepsilon \cdot n^2$ " "

Should be easier to detect

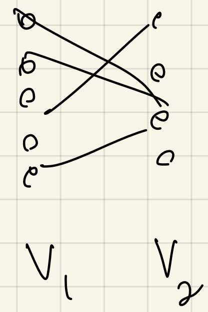


Bipartiteness:

- Can color nodes red/blue s.t. no edge monochromatic
- Can partition nodes into (V_1, V_2) s.t.

$$\underbrace{\nexists\ \underset{(u,v)}{\overset{\shortparallel}{e}} \in E \quad \text{s.t.} \quad u,v \in V_1 \text{ or } u,v \in V_2}_{\text{not bipartite } \Rightarrow\ \forall (V_1,V_2)\ \exists \text{ "violating edge"}} \Big\} \text{ "violating edges"}$$

equivalent definitions

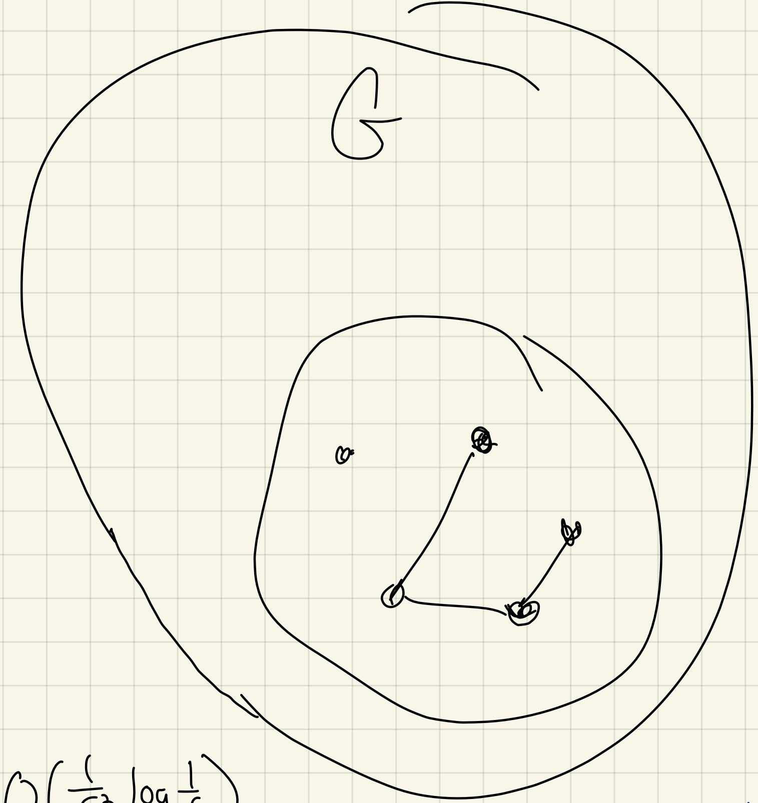


ε-far from bipartite: (definition)

- must remove $> \varepsilon \cdot n^2$ edges to make bipartite
- $\forall$ partitions $V = (V_1, V_2)$, $> \varepsilon \cdot n^2$ violating edges

equivalent

Testing Algorithms:



- Testing exact bipartiteness:

  e.g. BFS

- Proposed testing algorithm:

  - Pick sample of nodes of size $O\left(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon}\right)$
  - Consider induced graph on sample ← ignore nodes not in sample; ignore edge s.t. ≥ 1 endpt is not in sample
  - If bipartite, output PASS ← e.g. BFS

    else output FAIL

This actually works!!

A first attempt at a proof?

if G bipartite, induced graph is bipartite, so algorithm passes ✓

if G ε-far from bipartite:

must remove εn^2 edges to make it bipartite

equivalently:

$\forall$ partition V_1, V_2 have $> \varepsilon n^2$ violating edges ($> \varepsilon$ fraction of slots in adj matrix)

$\Rightarrow$ $\forall (V_1, V_2)$ a sample of edges of size $\geq \Theta\left(\frac{1}{\varepsilon} \log \frac{1}{\delta}\right)$

hits a (V_1, V_2)-violating edge with prob $\geq 1 - (1-\varepsilon)^{\frac{c}{\varepsilon}\log\frac{1}{\delta}}$

$\geq 1 - e^{-c \cdot \log \frac{1}{\delta}} = 1 - e^{-\log 1/\delta} = 1 - \delta$

(set $c = 1$)

Great! ?

need to hit violating edge for every partition

how is this an algorithm?

no edge violates all partitions

Lets try to use the "partition" defn of bipartiteness:

<u>Algorithm 0</u> (horrible runtime, but maybe ok query complexity?)

Pick $m = \Theta(?)$ random edge slots & query

$\forall$ partitions (V_1, V_2):

$\text{violating}_{(V_1,V_2)} \leftarrow$ # violating edges in sample wrt (V_1, V_2)

If $\forall (V_1, V_2)$ $\text{violating}_{(V_1,V_2)} > 0$ then output FAIL

else output PASS

Wait! How small should δ be?

Recall: All partitions are bad

But: if any partition "looks" good, the algorithm outputs PASS

Probability any partition "looks" good:

for one partition (V_1, V_2), $\Pr[(V_1, V_2) \text{ looks good}] \geq 1-\delta$

for all partitions (V_1, V_2), $\Pr[\text{all } (V_1, V_2) \text{ look good}] \geq 1 - \underbrace{2^n \cdot \delta}_{\text{union bnd over } 2^n \text{ partitions}}$

$\therefore$ need $\delta \ll \frac{1}{2^n}$?

would imply $m = \Theta\left(\frac{1}{\varepsilon} \log \frac{1}{\delta}\right) = \Theta\left(\frac{1}{\varepsilon} \log 2^n\right) = \underbrace{\Theta\left(\frac{n}{\varepsilon}\right)}_{}$

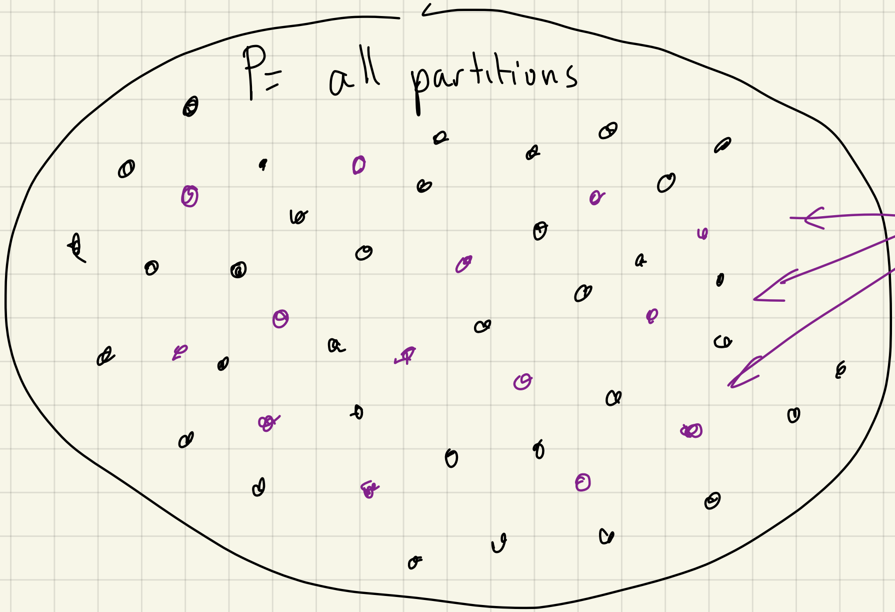
runtime is $\Theta(m^2)$
so not sublinear

Do we really need a union bnd?

Do we need to try all partitions?

(or can we find few "representative" partitions that are close to all partitions?)

Plan: Consider small set of representatives



$|P| = 2^n$

R = purple points are "representatives"

- $R \subseteq P$
- every member of P is close to member of R

Useful R satisfies:

$$\forall\ p \in P \quad \exists\ r \in R \quad \text{s.t.} \quad \text{dist}(p, r) \leq \varepsilon$$

Hope: (1) testing R tells you something about P

(2) $|R| \ll |P|$

(so union bound isn't as bad)

(1) if $p \in P$ has property then $\exists\ r \in R$ which is close to having property

(2) if $\forall\ p \in P$, p is far from having property then all r are far

Plan :

Find "representative" partitions s.t. all partitions are $\frac{\varepsilon}{2}$-close to some representative.

- if G ε-far from bipartite then $\forall$ partitions $\geq \varepsilon n^2$ violating edges

  so $\forall$ representative partitions $\geq \varepsilon n^2 - \frac{\varepsilon}{2} n^2$ violating edges
  $= \frac{\varepsilon}{2} n^2$

- if G bipartite then $\exists$ partition with 0 violating edges

  so $\exists$ representative partition with $< \varepsilon n^2$ violating edges

Algorithm 1

1. pick U, U' randomly from V

$|U| = \Theta(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ nodes — used to define set of partitions

$|U'| = \Theta(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$ nodes — used to define random edges for testing partitions:

pair off $U' = \{u_1, v_1, u_2, v_2, \ldots\}$

to $P = \{(u_1,v_1)(u_2,v_2)\ldots\}$ pairs

If U not bipartite, FAIL

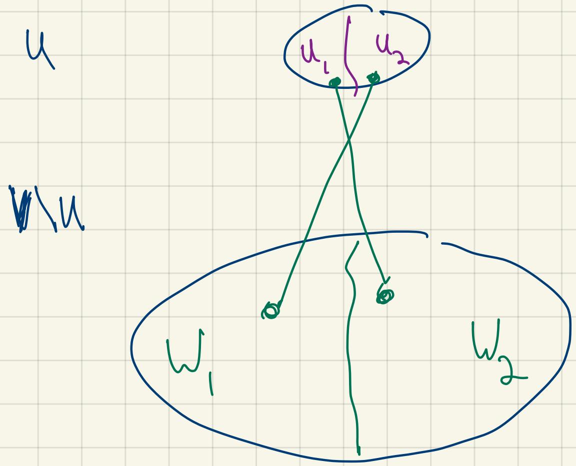
2. $\forall$ partitions of U into U_1, U_2:

consider only $2^{|U|}$ partitions?

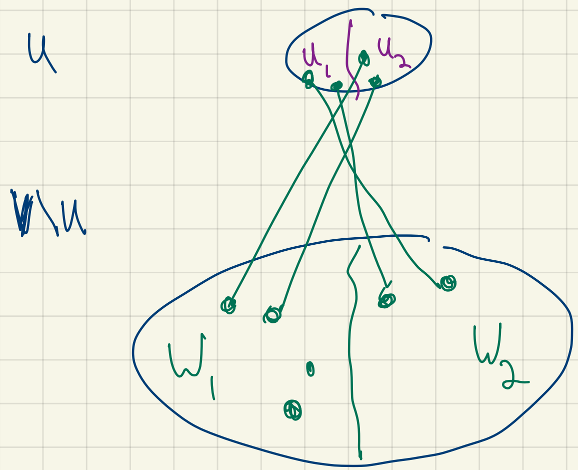
- define oracle (see below) which partitions graph into Z_1, Z_2 based on U_1, U_2
- $\forall\ u \in U'$ call oracle to see if u in Z_1 or Z_2
- Count $\#\{(u,v) \in P$ that violate $Z_1, Z_2\}$
  if fraction $\leq 3/4 \cdot \varepsilon$ output PASS + halt
  else continue to next partition

why pass if see any violations? since we don't check all partitions, might miss the perfect one

3. FAIL



Given partition of U into U_1, U_2, define ORACLE
to partition whole graph:



Query: node v
Oracle answer: Z_1 or Z_2 or "bad partition"

$Z_1 = U_1 \cup W_1$
$Z_2 = U_2 \cup W_2$

Oracle algorithm:

output Z_1 if
- $v \in U_1$
- v has nbr in U_2 but not U_1
- v has no nbr in U_1 or U_2

else output Z_2 if
- $v \in U_2$
- v has nbr in U_1 but not U_2

else output "bad partition"

we get here only if v has nbr in both U_1 + U_2

Runtime: $O(|U|)$ per query

Algorithm 1

1. pick U, U' randomly from V

$U = \Theta(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ nodes — used to define set of partitions

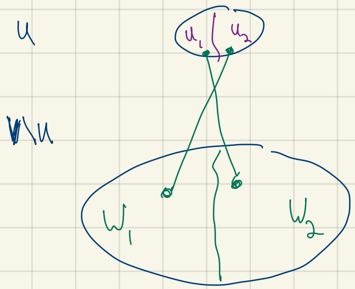
$U' = \Theta(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$ nodes — used to define random edges: pair off $U' = \{u_1, v_1, u_2, v_2, \ldots\}$ to $P = \{(u_1,v_1), (u_2,v_2), \ldots\}$ pairs

If U not bipartite, FAIL

2. $\forall$ partitions of U into U_1, U_2:

- define oracle (see below) which partitions graph into Z_1, Z_2 based on U_1, U_2
- $\forall u \in U'$ call oracle to see if u in Z_1 or Z_2
- count $\#\{(u,v) \in P$ that violate $Z_1, Z_2\}$
  if fraction $\leq 3/4\,\varepsilon$ output PASS + halt
  else continue to next partition

3. FAIL



Query Complexity:

$O(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$

Time Complexity:

$O(2^{\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}})$

no dependence on n

can improve dependence on ε

Behavior: need to show that if G bipartite, likely to pass
+ if G ε-far from bipartite, likely to Fail

if G is ε-far:

all partitions Z_1, Z_2, including those tested by algorithm, have $> \varepsilon n^2$ violating edges

$\forall Z_1, Z_2$ $\Pr[\text{fraction of violating edges in } P \text{ is } \leq \frac{3}{4}\varepsilon n^2] << \frac{1}{8 \cdot 2^{|U|}}$

$\Pr[\text{PASS}] = \Pr[\text{any } Z_1, Z_2 \text{ passes}] \leq 2^{|U|} \cdot \frac{1}{8 \cdot 2^{|U|}} \leq \frac{1}{8}$

Algorithm 1

1. pick U, U' randomly from V

   $U = \Theta(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ nodes — used to define set of partitions

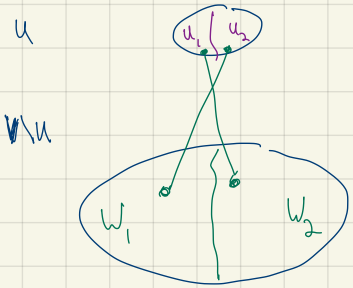
   $U' = \Theta(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$ nodes — used to define random edges: pair off $U' = \{u_1, v_1, u_2, v_2, \dots\}$ to $P = \{(u_1,v_1), (u_2,v_2), \dots\}$ pairs

   If U not bipartite, FAIL

2. $\forall$ partitions of U into U_1, U_2:

   - define oracle (see below) which partitions graph into Z_1, Z_2 based on U_1, U_2
   - $\forall\, u \in U'$ call oracle to see if u in Z_1 or Z_2
   - count $\#\{(u,v) \in P$ that violate $Z_1, Z_2\}$ if fraction $\leq 3/4\,\varepsilon$ output PASS + halt else continue to next partition

3. FAIL



if G is bipartite:

does it pass?

Let (Y_1, Y_2) be bipartite partition.

$\#$ violating edges $= 0$

For sample U, partition according to Y_1, Y_2:

$U_1 \leftarrow U \cap Y_1$

$U_2 \leftarrow U \cap Y_2$

$\}$ partition of U

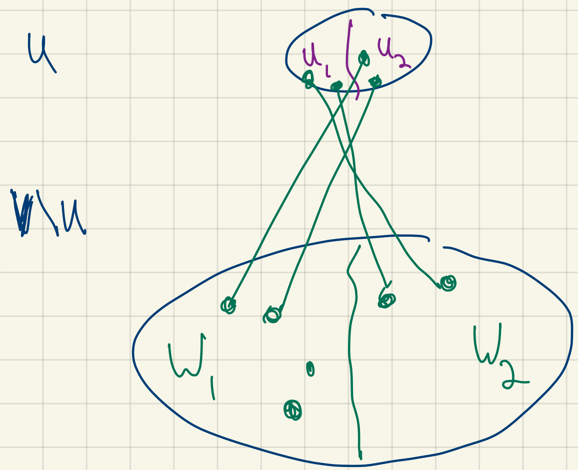
Use U_1, U_2 to partition V: $W_1^{U_1 U_2}$, $W_2^{U_1 U_2}$

only a thought process. (algorithm only partitions U')

Question: how similar is (Y_1, Y_2) to $(W_1^{U_1, U_2}, W_2^{U_1, U_2})$?

how many extra violating edges?

Given partition of U into U_1, U_2, define ORACLE to partition whole graph:



Query: node v

Oracle answer: Z_1 or Z_2 or "bad partition"

Oracle algorithm:

output Z_1 if

$v \in U_1$

v has nbr in U_2 but not U_1

v has no nbr in U_1 or U_2

only place where "oracle" has a "choice" + can do something different than Y_1, Y_2

else output Z_2 if

$v \in U_2$

v has nbr in U_1 but not U_2

$Z_1 = U_1 \cup W_1$

$Z_2 = U_2 \cup W_2$

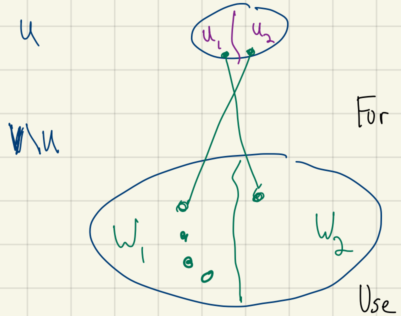
else output "bad partition"

we get here only if v has nbr in both U_1 + U_2

Runtime: $O(|U|)$ per query

$\forall$ partitions of u into u_1, u_2:

- induce partition $(\underbrace{u_1 \cup W_1}^{Z_1}, \underbrace{u_2 \cup W_2}^{Z_2})$ on whole graph:



Let (Y_1, Y_2) be bipartite partition.

\# violating edges $= 0$

For sample u, partition according to Y_1, Y_2:

$u_1 \leftarrow u \cap Y_1$

$u_2 \leftarrow u \cap Y_2$

Use u_1, u_2 to partition V: $W_1^{u_1 u_2}$, $W_2^{u_1 u_2}$

\# violating edges in $(W_1^{u_1 u_2}, W_2^{u_1 u_2})$:

$$\le \underbrace{0}_{\#\text{ violating edges in } (Y_1, Y_2)} + \#\text{ edges adjacent to any } v \underbrace{\text{with no nbr in } u}_{\text{divide into two cases!}}$$

$A = \{v \text{ s.t. } \deg(v) < \frac{\varepsilon}{4} n\}$ "small degree"

$B = \{v \text{ s.t. } \deg(v) \ge \frac{\varepsilon}{4} n\}$ "high degree"

$$\le \underbrace{\frac{\varepsilon n}{4}}_{\text{max deg of } v \in A} \cdot \underbrace{n}_{\text{upper bnd on } |A|} + \underbrace{n}_{\text{max deg of } v \in B} \cdot \underbrace{\square}_{\text{upper bnd on } |B|}$$

So need to upper bound $|B|$: nodes with high degree that don't neighbor u

recall: U is random sample, $|U| = \Theta(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$

$B_U = \{ v$ s.t. $\deg(v) \geq \frac{\varepsilon}{4}n$ + v has no nbr in $U \}$

<u>Lemma</u> $\Pr_U\left[|B_U| \leq \frac{\varepsilon}{4}n\right] \geq 7/8$

<u>Pf</u> $\forall v$ of degree $\geq \frac{\varepsilon}{4}n$ set $\sigma_v \leftarrow \begin{cases} 1 & \text{if } v \in B \\ 0 & \text{o.w.} \end{cases}$

has no nbr in U

$$E[\sigma_v] = \Pr[\sigma_v = 1]$$

$$= \left(\Pr[i^{th} \text{ node of } U \text{ isn't nbr of } v]\right)^{|U|}$$

$$\leq \left(1 - \frac{\varepsilon}{4}\right)^{|U|} \leq \left(1 - \frac{\varepsilon}{4}\right)^{\Theta(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})} \leq \frac{\varepsilon}{32}$$

uses high degree of v

choose to be $\frac{4}{\varepsilon}\cdot\log{}^{32}/\varepsilon$

$$E\Big[\sum_{\substack{v \text{ s.t.} \\ \deg(v) \geq \frac{\varepsilon}{4}n}} \sigma_v\Big] \leq \frac{\varepsilon n}{32} \quad \text{so} \quad \Pr\Big[\sum \sigma_v \geq \underbrace{\frac{8 \cdot \varepsilon n}{32}}_{\frac{\varepsilon n}{4}}\Big] \leq \frac{1}{8}$$

by Markov's ≠ ∎

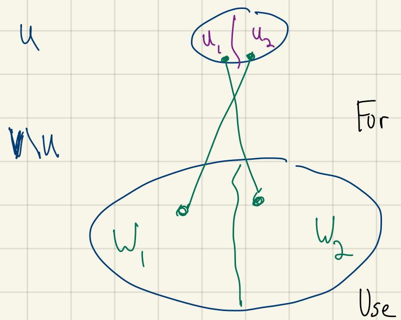
$\forall$ partitions of U into U_1, U_2:

- induce partition $\overset{(Z_1,\ Z_2)}{(U_1 \cup W_1,\ U_2 \cup W_2)}$ on whole graph:

  output Z_1 if
  - $v \in U_1$
  - v has nbr in U_2 but not U_1
  - v has no nbr in U_1 or U_2

  else output Z_2 if
  - $v \in U_2$
  - v has nbr in U_1 but not U_2

  else output "bad partition"



Let (Y_1, Y_2) be bipartite partition.

\# violating edges $= 0$

For sample U, partition according to Y_1, Y_2:

$U_1 \leftarrow U \cap Y_1$

$U_2 \leftarrow U \cap Y_2$

Use U_1, U_2 to partition V: $W_1^{U_1 U_2}$, $W_2^{U_1 U_2}$

\# violating edges in $(W_1^{U_1U_2}, W_2^{U_1U_2})$:

$$\leq \underbrace{0}_{\#\text{ violating edges in } (Y_1, Y_2)} + \overbrace{\#\text{ edges adjacent}}^{\#\text{edges that can differ between } (Y_1,Y_2) \text{ \& } (W_1^{U_1U_2}, W_2^{U_1U_2})} \underbrace{\text{to any } v \text{ with no nbr in } U}_{\text{divide into two cases!}}$$

$A = \{v \text{ s.t. } \deg(v) < \frac{\varepsilon}{4} n\}$ "small degree"

$B = \{v \text{ s.t. } \deg(v) \geq \frac{\varepsilon}{4} n\}$ "high degree"

$$\leq \underbrace{\frac{\varepsilon \cdot n}{4}}_{\text{max deg of } v \in A} \cdot \underbrace{n}_{\text{upper bnd on } |A|} + \underbrace{n}_{\text{max deg of } v \in B} \cdot \underbrace{\frac{\varepsilon \cdot n}{4}}_{\text{upper bnd on } |B|} \leq \frac{\varepsilon n^2}{2}$$

(with prob $\geq 7/8$)

$\Rightarrow E[\text{fraction of } (u,v) \in P \text{ violating } W_1^{U_1U_2}\, W_2^{U_1U_2}] \leq \varepsilon/2$

so $\Pr[$ " " " " " $\geq \frac{3\varepsilon}{4}] << 1/8$

use Chernoff bnds + samples to show this recall: U is random sample, $|U| = \Theta(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$

$B_u = \{v$ s.t. $\deg(v) \geq \frac{\varepsilon}{4}n$ & v has no nbr in $U\}$

$$\Rightarrow \Pr[\text{output fail}]$$

$$\leq \underbrace{\Pr_u[\text{output fail} \mid |B_u| > \tfrac{\varepsilon}{4}n]}_{\leq 1} \cdot \underbrace{\Pr_u[|B_u| > \tfrac{\varepsilon}{4}n]}_{\leq 1/8}$$

$$+ \underbrace{\Pr[\text{output fail} \mid |B_u| \leq \tfrac{\varepsilon}{4}n]}_{\leq 1/8} \cdot \underbrace{\Pr_u[|B_u| \leq \tfrac{\varepsilon}{4}\cdot n]}_{\leq 1}$$

$$\leq \frac{1}{8} + \frac{1}{8} = \frac{1}{4}$$

Comments

Can improve runtime to poly$(1/\varepsilon)$

Proposed testing algorithm actually works

In adjacency list model (sparse graphs) need

$\Omega(\sqrt{n})$ queries,

Why more?

finer grain distinction

dense model: bipartite vs. $\varepsilon \cdot n^2$ edges need to be removed

sparse model: bipartite vs. $\varepsilon \cdot \Delta \cdot n$ edges need to be r

Other problems: Partition Properties

e.g. Max cut

can we partition into k $\approx$ equal sized groups $1 \ldots k$ s.t. fraction of edges between groups i & j $\approx a_{ij}$

inputs to problem

similar oracle-based algorithms:

<u>Maxcut</u>: pick random sample S

$\forall$ partitions of S into (S_1, S_2), create oracle:

if $v \in V \setminus S$ has more edges to S_2 than S_1

put in W_1

else " " W_2

then estimate # edges between $(W_1 \cup S_1)$ & $(W_2 \cup S_2)$

Output max value

Algorithm 1

1. pick U, U' randomly from V

$U = \Theta(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ nodes

used to define set of partitions

$U' = \Theta(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$ nodes

used to define random edges for testing partitions:

pair off $U' = \{u_1, v_1, u_2, v_2, \dots\}$

to $P = \{(u_1,v_1)(u_2,v_2)\dots\}$ pairs

If U not bipartite, FAIL

2. $\forall$ partitions of U into U_1, U_2:

Consider only $2^{|U|}$ partitions?

- induce partition $(U_1 \cup W_1, U_2 \cup W_2) = (Z_1, Z_2)$ on whole graph:

$O(|U|)$ time Subroutine which only call for $v \in U'$

Partition: $\forall v \in V$ (including $v \in U$)

- if v has nbr in U_2, put in W_1
- if v " " " U_1, " " W_2
- " " " " " both $\Rightarrow$ bad partition! Continue to next partition
- " " " " " neither, put in W_1

- Count $\#\{(u,v) \in P$ s.t. violate $(Z_1, Z_2)\}$

  if fraction $\leq 3/4\,\varepsilon$ PASS ~~halt~~

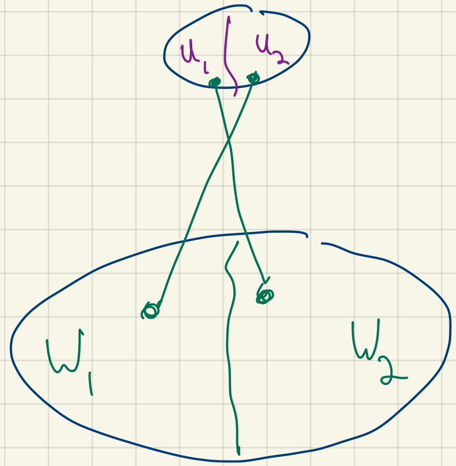
  else continue to next partition

why pass if see any violations? since we don't check all partitions, might miss the perfect one

3. FAIL

U

$V \setminus U$



Algorithm 1

1. pick U, U' randomly from V

$U = \Theta\left(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}\right)$ nodes

used to define set of partitions

$U' = \Theta\left(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon}\right)$ nodes

used to define random edges for testing partitions: pair off $U' = \{u_1, v_1, u_2, v_2, \ldots\}$ to $P = \{(u_1,v_1)(u_2,v_2)\ldots\}$ pairs

If U not bipartite, FAIL

2. $\forall$ partitions of U into U_1, U_2:

Consider only $2^{|U|}$ partitions?

- induce partition $(U_1 \cup W_1, U_2 \cup W_2)$ on whole graph:

(Z_1, Z_2)

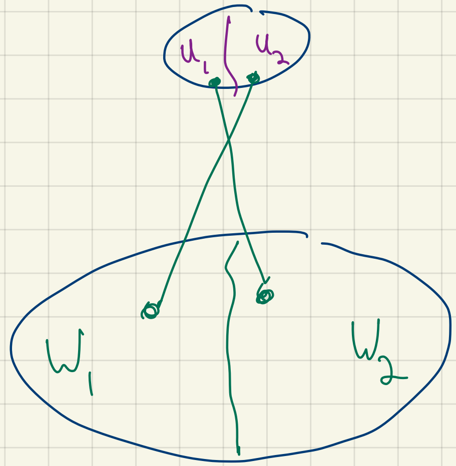
$O(|U|)$ time subroutine which only call for $v \in U'$

Partition: $\forall v \in V$ (including $v \in U$)

- if v has nbr in U_2, put in W_1
- if v " " " U_1, " " W_2
- " " " " " both $\Rightarrow$ bad partition! Continue to next partition
- " " " " " neither, put in W_1

U

$V \setminus U$



- count $\#\{(u,v) \in P$ s.t. violate $(Z_1, Z_2)\}$

  if fraction $\leq 3/4\ \varepsilon$ PASS ~~+~~halt

  else continue to next partition

why pass if see any violations? since we don't check all partitions, might miss the perfect one

3. FAIL