CS314 - Approximate Vertex Cover

Note Title 4/9/2010

Announcements

- HW due next Wednesday

Definition:

Let OPT(X) be the value of an optimal solution, + A(X) be the value of an algorithm A's solution (on input X).

A is an <u>$\alpha(n)$-approximation</u> if + only if

→ $$\frac{OPT(X)}{A(X)} \le \alpha(n) \quad \text{and} \quad \frac{A(X)}{OPT(X)} \le \alpha(n)$$

for all inputs X of size n.

Last time: $\frac{3}{2}$-approx

Usually, only one of these inequalities is important.

$$\underbrace{\frac{OPT(X)}{A(X)} \leq \alpha(n)}_{\text{minimization problems}} \quad \text{and} \quad \underbrace{\frac{A(X)}{OPT(X)} \leq \alpha(n)}_{\text{maximization problems}}$$

→ minimization problems

→ maximization problems

Last time:

We found a schedule where $T \le \frac{3}{2} T^*$.

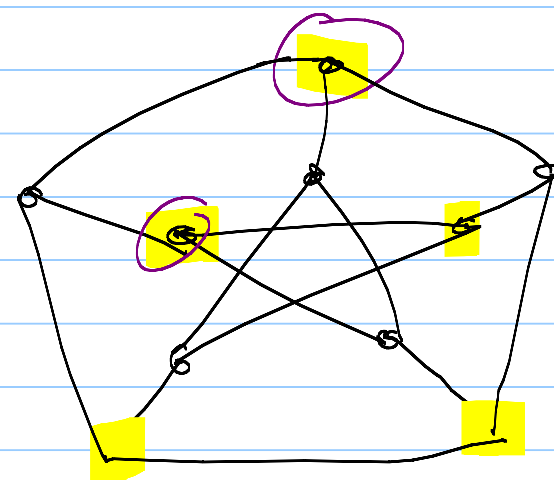
So $\frac{T}{T^*} \le \frac{3}{2}$ and $\frac{T^*}{T} \le \frac{2}{3}$.

This means we had a $\frac{3}{2}$-approximation.

(we wanted to minimize T's value)

Vertex Cover

- A set of vertices which "covers" every edge in the graph.



Note: this will be minimization problem - want V.C. as small as possible.

What's a natural approximation method?

greedy

pick someone with a large #
of uncovered edges

Pseudocode:

```
GreedyVertexCover(G):
    C ← ∅
    while G has at least one edge
        v ← vertex in G with maximum degree
        G ← G \ v
        C ← C ∪ v
    return C
```

Thm: Greedy Vertex Cover gives an $O(\log n)$ - approximation.

Pf:

Notation: Let G_i be the graph after i loop iterations, and d_i be maximum degree in G_{i-1}.

Also: C^* is optimal cover, & $|E(G_i)|$ = # edges in G_i.

pf (cont):

```
GreedyVertexCover(G):
    C ← ∅
    G_0 ← G
    i ← 0
    while G_i has at least one edge
        i ← i + 1
        v_i ← vertex in G_{i−1} with maximum degree
        d_i ← deg_{G_{i−1}}(v_i)
        G_i ← G_{i−1} \ v_i
        C ← C ∪ v_i
    return C
```

more proof: C^* is also a vertex cover for G_{i-1}, so

$$\sum_{v \in C^*} \deg_{G_{i-1}}(v) \geq |E(G_{i-1})|$$

$\Rightarrow$ average degree in G_{i-1} of a vertex in C^*

$$\geq \frac{|E(G_{i-1})|}{|C^*|}$$

☆ $\Rightarrow$ $d_i \geq \dfrac{|E(G_{i-1})|}{|C^*|}$ (since d_i is maximum degree)

For any $j \geq i-1$, G_j has fewer edges than G_{i-1},

so $d_i \geq \dfrac{|E(G_j)|}{|C^*|}$ ☆

$$OPT = |C^*|$$

$$\sum_{i=1}^{OPT} d_i \geq \sum_{i=1}^{OPT} \frac{|G_{i-1}|}{OPT} \geq \sum_{i=1}^{OPT} \frac{|E(G_{OPT})|}{OPT}$$

fill in ☆ fill in ☆

and $$\sum_{i=1}^{OPT} \frac{|E(G_{OPT})|}{OPT} = |E(G_{OPT})| = |E(G)| - \sum_{i=1}^{OPT} d_i$$

Rewrite: $$\sum_{i=1}^{OPT} d_i \geq |E(G)| - \sum_{i=1}^{OPT} d_i$$

$$2\left(\sum_{i=1}^{OPT} d_i\right) \geq |E(G)| \Rightarrow \sum_{i=1}^{OPT} d_i \geq \frac{|E(G)|}{2}$$

$$\sum_{i=1}^{OPT} d_i \geq \frac{|E(G)|}{2}$$

So I've deleted half the edges in OPT repetitions of the loop.

So after $OPT(\lg m)) \leq 2\, OPT \lg n$ iterations, all edges are gone.

Unfortunately, this can't be improved.

There is a nice (recursive) construction of a graph of size n for which greedy returns a vertex cover of size $\Omega(OPT \cdot \log n)$.

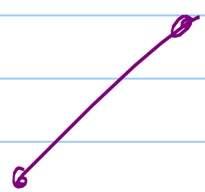
Don't always be greedy!

Another idea –

take any edge $e = (u, v)$ in G.

What **must** be in any vertex cover?

either u or v is in C^*



so add both

Pseudocode

$\textsc{DumbVertexCover}(G)$:

$\quad C \leftarrow \varnothing$

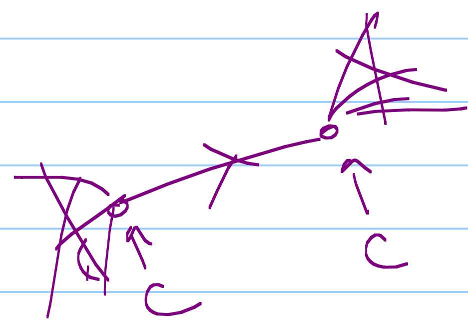
$\quad$ while G has at least one edge

$\qquad (u, v) \leftarrow$ any edge in G

$\qquad G \leftarrow G \setminus \{u, v\}$

$\qquad C \leftarrow C \cup \{u, v\}$

$\quad$ return C



Dumb Vertex Cover is a 2-approximation!

C^* contains one of the 2 vertices I add in every loop iteration.

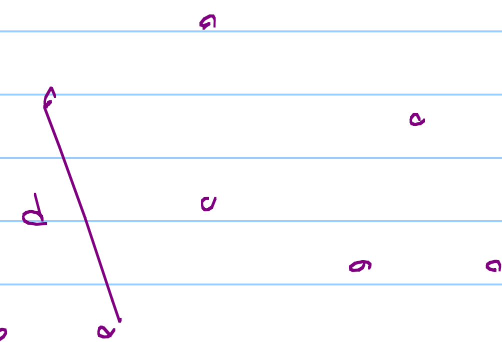
$$\Rightarrow |C^*| \leq \frac{1}{2}|C|$$

Next time: Traveling Salesman

Q: Is there a Hamiltonian cycle in a weighted (complete) graph with length $\leq k$?

NP-Hard. Why?

Hand you G & ask if there is a Ham cycle.



Take G + make all edges of weight $= 1$.

Any other edge, give larger weight say $= \cancel{2}$. n

Ask: is there a T.S. tour of length $\leq n$?