session 1

Network Design Course

Date FALL 2011

Today: of course not every detail of these problems

1- Set Cover: lots of Applications, e.g., in wireless network: minimum set of transmitter covering every area

2- maximum Coverage: wireless networks covering, also circuit layout, job scheduling, facility location, etc.

3- Unique Coverage: several applications, e.g., maximize the area of interference free in wireless network

1- Set cover: a Universe U of n elements, a collection X of subsets of U, $S = \{S_1, \dots, S_k\}$, and a cost function $c: S \to Q^+$

find a minimum cost subcollection X' of S that cover all elements of U

frequency of an element is the number of sets it is in: say F is the maximum frequency.

Thm 1: There is a minimum of $O(\lg n)$ and F approximation for set-cover.

A greedy $O(\lg n)$ approximation:

1. $C \leftarrow \emptyset$
2. while $C \neq U$ do

   find the set S with the most cost-effectiveness; i.e. minimizes $\alpha = \frac{cost(S)}{|S-C|}$

   Pick S, and for each $e \in S - C$, set $price(e) = \alpha$ only for the analysis (cost(S) is amortized to its new covered elements) thus: $\boxed{cost(\text{Greedy ALG}) \leq \sum_{e \in U} price(e)}$

   $C \leftarrow C \cup S$
3. output the picked sets.

Analysis: OPT covers every thing (including $\bar{C}$) since there should be one set with cost-efficiency $\frac{OPT}{|U-C|}$. Number the elements of U in the order in which we add them to C (break ties arbitrarily)

(why: remove C from opt and U

$cost(OPT) = cost(S_1) + cost(S_2) + \dots cost(S_k) \geq \frac{cost(S_1)}{|S_1|}|S_1| + \dots + \frac{cost(S_k)}{|S_k|}|S_k|$

$\geq \alpha |S_1| + \alpha |S_2| + \dots + \alpha |S_k| \geq \alpha |U-C|$. Thus $\frac{cost(opt)}{|U-C|} \geq \alpha$).

Thus

$\{cost(\text{of } \underset{Greedy}{ALG}) = \sum_{e \in U} Price(e) = Price(e_1) + Price(e_2) + \dots + Price(e_n)$

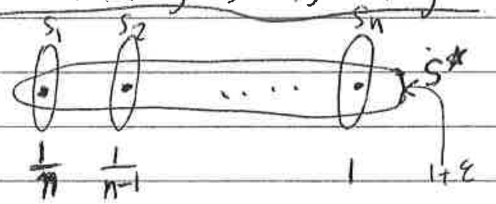
$\leq \frac{cost(opt)}{n-1+1} + \frac{cost(opt)}{n-2+1} + \dots + \frac{cost(opt)}{n-i+1} + \dots + \frac{cost(opt)}{n-n+1}$

(why, in the iteration in which element e_i is added $|U-C| \geq n-i+1$ by definition of e_i)

$e_1\ e_2\ \overbrace{e_3\ e_4\ e_5}^{S}\ \dots\ e_n$, C

$\leq cost(opt)\left(\frac{1}{n} + \frac{1}{n-1} + \dots + 1\right) = cost(opt)\ H_n = cost(opt)\ \ln n$

The analysis is tight:

costs: 

opt choose s^* while the greedy picks $s_1, s_2, \dots, s_n$. The cost of opt is $1+\varepsilon$ while the cost of Greedy is $H_n = \ln n$

A greedy F approximation:

1. Cover A $\leftarrow \emptyset$
2. while E contains elements e not covered by Cover A
 add all sets containing e to Cover A.
 add e to U′ (initially U′ is empty) — just for the sake of analysis

Analysis: U' is an independent set; i.e. no element of our collection X can cover both of them. Thus opt should ~~ever~~ need at least $|U'|$ set. We have at most $F|U'|$ sets since each element appears at most F times. Thus we have an F-approximation here.
It is easy to construct a tight example (exercise)

Doese the above analysis work for arbitrary cost of sets?

Answer NO: only when all ~~weights~~ costs are ones.

However still we can obtain F approximation by other techniques such as linear programming. How? see below

$$obj = \text{minimize} \sum_{j=1}^{|X|} cost(x_j)$$

$$\text{s.t.} \quad \sum_{i \in X_j} x_j \geq 1 \qquad i = 1 \ldots |U|$$

$$x_i \geq 0.$$

Consider an element i, at least one of x_j where $i \in X_j$ is at least $\frac{1}{F}$ choose that set and all of its elements and iterate!!!

We pay at most $F \, obj \leq F \, OPT$ (linear program vs. integer program).

Using LP rounding we can also obtain $O(\ln n)$ approximation, as before.

However Feige showed

Thm: There is no $(1-\varepsilon)\ln n$ - approximation to set cover unless $NP \subseteq DTIME(n^{\log\log n})$. (essentially no improvement even by a constant factor)

The problem is equivalent to MINIMUM Dominating set in GRAPHS

See section 2 and 13 of Vazirani for more details and extensions.