CS 538: Practice: Sample Problems

Section: MW 2-3:15 pm

Instructions:
1. I leave plenty of space on each page for your computation. If you need more sheet, please attach your work right behind the corresponding problem. If your answer is incorrect but you show the computation process, then partial credits will be given. Please staple your solution and use the space wisely.

2. Full credit will be given only to the correct solution which is described clearly. Convoluted and obtuse descriptions might receive low marks, even when they are correct. Also, aim for concise solutions, as it will save you time spent on write-ups, and also help you conceptualize the key idea of the problem.

First Name:

Last Name:

Group ID:

Score: /

Problem 1 Weighted Vertex Cover

In the linear-time 2-appproximation of weighted vertex cover algorithm for a graph $G(V, E)$, , we introduced a new set of variables p_v which reprsent how much we are willing to pay in order to put vertex v in the vertex cover S; at the end, if $p_v = c(v)$ where $c(v)$ is the weight of vertex v, then the vertex v is selected. If $p_v < c(v)$, then we will not going to use vertex v. The algorithm is as

- Input: undirected, unweighted, graph $G = (V, E)$
- $\mathbf{p} = (0, \cdots, 0)$
- $\mathbf{y} = (0, \cdots, 0)$
- for each edge $(u, v) \in E$
  - if $p_u < c(u)$ and $p_v < c(v)$ then
    - $y_{(u,v)} := \min\{c(u) - p_u, c(v) - p_v\}$
    - $p_u := p_u + \min\{c(u) - p_u, c(v) - p_v\}$
    - $p_v := p_v + \min\{c(u) - p_u, c(v) - p_v\}$
- $S := \{v : p_v \geq c(v)\}$
- return $S, \mathbf{y}$

(a) Why $cost(S) = \sum_{v \in S} c(v) \leq \sum_{v \in V} p_v$

(b) Why $\sum_{v \in V} p_v = 2 \sum_{(u,v) \in E} y_{(u,v)}$

Problem 2 Definition

(a) What is the difference between Minimal Spanning Tree and Steiner Tree?
(b) What is the difference betwee Steiner Tree and Metric Steiner Tree?
(c) What is a perfect matching problem ?
(d) What is a vertex cover problem?
(e) What is a set cover problem?
(f) What is a network max flow cut problem?

Problem 3 Graph: 10pts

Please show that for any arbitrary graph $G = (V, E)$ there is an even number of nodes with an odd degree of edges.

Problem 4 Vertex Cover: 10pts

Suppose we are given an algorithm listed below. Does this algorithm guarantee a 2OPT? If yes, explain why; if not, please give an counter example.

Algorithm 1 Vertex Cover Made Easy

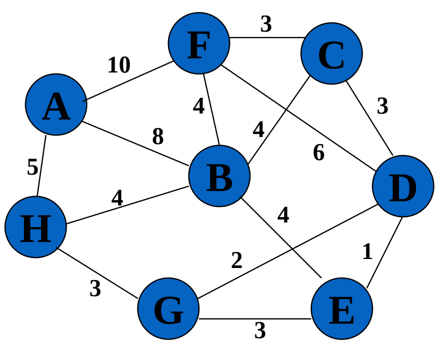
Require: Graph $G = (V, E)$
Ensure: Output the vertex cover of 2OPT
  Start of algorithm
  Vertex cover $C := \emptyset$
  while $\exists (u, v) \in E$ such that $u \notin C$ and $v \notin C$ **do**
    $C := C \cup \{u\}$
  Return C
  End of algorithm

Problem 5 Steiner Tree: 20pts

Please briefly describe a 2OPT algorithm for a steiner tree problem (X, d) where $X = R \cup S$ (R is the set of required vertices and S is the set of optional vertices) and d is the distance function that DOES NOT have to satisfy triangle inequality.

Problem 6 Miminal Spanning Tree: Krusal

On the next page, we are given a graph G. Please execute Kruskal's algorithm and find the MST(G) and its weight.



Sort the edges by increasing edge weight

edge	d_v	
(D,E)	1	
(D,G)	2	
(E,G)	3	
(C,D)	3	
(G,H)	3	
(C,F)	3	
(B,C)	4	

edge	d_v	
(B,E)	4	
(B,F)	4	
(B,H)	4	
(A,H)	5	
(D,F)	6	
(A,B)	8	
(A,F)	10	

Problem 7 Travelling Salesman Problem : G_TSP_R ← M_TSP_R

(a) Please describe (in order) the 5 steps algorithms we discussed in class in order to solve an M_TSP_R problem with a 1.5 OPT.

(a-1)

(a-2)

(a-3)

(a-4)

(a-5)

(b) In the M_TSP_R algorithm, let $O :=$ the set of vertices that have odd degrees in MST T. Why finding the perfect maching (O, M) and adding it back to the T would guarantee an Eulerian cycle in graph $(T + M)$?

(c) Briefly describe how you would extend this implmentation to any arbitrary nonnegative symmetric distance funciton such that you can can easily find a 1.5 OPT of **G_TSP_R** ?

Problem 8 Min $\leftrightarrow$ Max: Find Dual

Please convert the following problems from min (max) LP to max (min) LP
(A) **maximize**: $5x_1 + 7x_2 + 6x_3 + 2x_4$
subject to: (1) $2x_1 + x_2 + x_3 + 3x_4 \leq 5$ (2) $x_1 + 3x_2 + x_3 + 2x_4 \leq 5$ (3) $x_1, x_2, x_3, x_4 \geq 0$

(B) minimize: $3y_1 + 6y_2 + 7y_3 + 7y_4$
subject to: (1) $5y_1 + 2y_2 + y_3 + 2y_4 \geq 20$ (2) $y_1 + 3y_2 + 2y_3 + 2y_4 \geq 30$ (3) $y_1, y_2, y_3, y_4 \geq 0$

Problem 9 Harmonic Sum

In the Set Cover problem, we analyzed the complexity of a greedy algorithm that is bounded from above by $\ln n + O(1)$ by using Harmonic sum. Please show that the Harmonic sum $\sum_{i=1}^{n} \frac{1}{i}$ is bounded from above by $\lceil \log_2 n + 1 \rceil$

Problem 10 Chernoff Bound: Q.10.1 on P. 267: 20pts

Theorem 1. *Let $X_1, \cdots, X_m$ be indepdent and identically distributed indicator random variables, with $\mu = \mathbb{E}[X_i]$. If $m \geq (3\ln(2/\delta))/(\epsilon^2\mu)$ then*

$$Pr\big(|\frac{1}{m}\sum_{i=1}^{m} X_i - \mu| \geq \epsilon\mu\big) \leq \delta$$

Please prove the above theorem by applying Chernoff bound.

Problem 11 Application: DNF Counting 4*5 pts

In class we talk about two algorithms (see chapter 10 in the textbook) for estimating the number of solutions for a given disjunctive normal form F. Let us call the Naive approach Alg-I (sampling from the solution space) and the one that cleverly samples from the TRUE solutions space as Alg-II. Let us assume that F contains t clauses and n variables. Please answer the following:

(a) Why is Alg-I not a good sampling approach

(b) In Alg-II, why is $|S|/|U| \geq 1/t$? In what scenario we will have $|S|/|U| = 1/t$? And in what scenario we have $|S|/|U| = 1$?

(c) If F is a k-DNF, what is the value $|SC_i|/|U|$ Is it a fixed number for all i?

(d) What is the major difference in sample space between Alg-I and Alg-II and why Alg-II is better?

Problem 12 Application: DNF Alg-I: Q.10.4 on P. 267: 20pts

Suppose we have a class of instances of the DNF satifiability problem, each with $\alpha(n)$ satisfying truth assignments for some polynomial α. Suppose we apply the naive appraoch of sampling assginments and checking wether they satisfy the formula. Show that, after sampling $2^{n/2}$ assginments, the probability of finding even a single satifying assignment for a given instance is exponenitally samll in n. [Hint: Union bound]

Problem 13 FPRAS: : Q.10.3 on P. 267: 20pts

Show that the following alternative definition is equivalent to the definition of an FPRAS given in the chapter: A *fully polynomial randomized approximation scheme (FPRAS)* for a problem is a randomized algorithm for which given an input x and any parameter ϵ with $0 < \epsilon < 1$ the algorithm outputs and $(\epsilon, 1/4)$-approximation in time that is polynomial in $1/\epsilon$ and the size of the input x (Hint: To boost the probability of success from $3/4$ to $1 - \delta$ consider the median of several independent runs of the algorithm.)

Problem 14 Your Project

In your final project, you implement several techniques described in a paper. Please summarize the selling point of the paper (such as how many order of magnitude the complexity is reduced) you are implementing and briefly describe the killer trick used in the algorithm that boosts the performance.