Topics in Approximation Algorithms: Assignment 3

Sayan Bhattacharya   Parinya Chalermsook

The problems marked by * are optional. You may solve them for extra credits.

Problem 1: Defending your houses

In this problem, we are given an undirected tree $G = (V, E)$ together with source s and subset $T \subseteq V$ of terminals. Your job is to serve as an *earth defender* to contain the fire set out by the arsonist. The arsonist initially invades vertex s by setting fire on s at time step 0. The status of s becomes "burning". In each time step t, starting from $t = 1, 2, \ldots$ the defender moves first by choosing at most B vertices that are not burning and place one *troop* on each such vertex, whose status becomes "saved". Once saved, a vertex would never burn throughout the game. Then the fire moves from currently burning vertices to all their neighbors, setting the unsaved vertices on fire. In short, for each time step, the defender saves B vertices, and the fire spreads to immediate neighbors.

The game ends when the fire can no longer spread. We say that a vertex v *survives* if it does not burn when the game ends. Our goal is to ensure that all vertices in T survive, while minimizing B (the cost per time unit).

Formal Definition: Given a tree $G = (V, E)$, source s and subset $T \subseteq V$ that needs to survive. Our goal is to pick a collection of vertices $U_1, U_2, \ldots, U_n \subseteq V$ such that $|U_t| \leq B$. For each terminal $v \in T$, let $P_v = (s = u_0, u_1, \ldots, u_s = v)$ be the unique path from s to v. A terminal $v \in T$ survives if and only if there is some $t : 1 \leq t \leq s$ such that $u_t \in \bigcup_{t' \leq t} U_{t'}$. Our goal is to compute the solution $\{U_t\}$ such that all vertices in T survive, while minimizing B.

Suppose that we root the tree G at vertex s. Layers of the tree are well-defined where L_τ contains vertices at distance τ from s (so s is at layer 0), and we have $V(G) = \bigcup_{\tau=1}^{\ell} L_\tau$. We use ℓ to denote the index of the last layer.

(a) Argue that we can assume w.l.o.g. that the defender only saves vertices at layer t in time step t. That is, we can assume $U_\tau \subseteq L_\tau$ for all $\tau = 1, \ldots, \ell$.

(b) Using observation from part (a), we can write the following integer program. Let L_τ denotes the set of vertices in layer τ of the tree, and ℓ be the number of layers in G.

(IP)

$$
\begin{aligned}
\min \quad & B \\
\text{s.t.} \quad & \sum_{v \in L_\tau} x_v \leq B \text{ for all } \tau = 1, \ldots, \ell \\
& \sum_{u \in P_v} x_u \geq 1 \text{ for all } v \in T \\
& x_v \in \{0, 1\} \text{ for all } v \in V
\end{aligned}
$$

Argue that this integer program is a valid formulation of the problem, i.e. a solution for (IP) can be turned into the feasible solution for the problem and vice versa.

(c) Relax the integer program to linear program. Design a randomized $O(\log |T|)$ approximation algorithm by LP rounding. You will need a Chernoff bound for this.

(d) Suppose $U_1, \ldots, U_\ell$ be a feasible solution such that U_t is a collection of vertices saved at time t. An *amortized cost* of this solution is $\max_t \frac{\sum_{t' \leq t} |U_{t'}|}{t}$. Show that any feasible solution with amortized cost at most B can be turned into a feasible solution such that at most B vertices are saved in every time step.

(e) Give an $O(\log \ell)$ approximation. This is an improved approximation ratio when $\ell << |T|$. (Hint: random cut techniques and part (d))

(f) *Construct an integrality gap lower bound of 2. More precisely, show an input (G, s, T) such that integral solution needs 2 vertices per time step to secure T, while fractional solution only needs 1.

Problem 2: Cutting short paths

Recall that, a threshold rounding for an LP is a standard technique of transforming a fractional LP solution to an integral one. Given an LP solution x, the threshold rounding would pick some value $\tau \in (0, 1)$ and turn a solution x into an integral solution by $x'_i = 1$ if and only if $x_i \geq \tau$; otherwise $x'_i = 0$. (refer to section 1.3 in the book)

Story: *Given an unweighted, undirected input graph $G = (V, E)$ with source s (think of this as MPI building) and destination t, Sayan wants to travel from MPI building to his apartment (i.e. vertex t). There are many paths he can travel from s to t, but knowing algorithms well, he always commutes through short paths, say, paths of length at most k. Now, a student who hates his class wants to make sure he suffers walking a long distance, so the student plans to remove some edges from G such that all paths of length at most k disappear. Of course, if the student spends too much time poking fun of Sayan, he will lose his study hours and might fail exams, so he is interested in minimizing the number of edges removed.*

Formal Definition: Given an undirected graph $G = (V, E)$, source and sink s, t, and integer k, find a subset of edges $E' \subseteq E$ such that $G' = (V, E \setminus E')$ does not contain any path of length at most k from s to t. Our goal is to minimize $|E'|$.

(a) Let $\mathcal{P}_k$ be the set of all paths from s to t of length at most k. We can write the following LP relaxation for the problem. Let x_e be a variable indicating whether e is cut.

(LP)

$$\begin{aligned} \min \quad & \sum_{e \in E} x_e \\ \text{s.t.} \quad & \sum_{e \in P} x_e \geq 1 \text{ for all } P \in \mathcal{P}_k \\ & x_e \in [0, 1] \end{aligned}$$

Argue that this is a relaxation of the problem and that this LP is polynomial-time solvable by providing a separation oracle.

(b) Show a k approximation algorithm based on threshold rounding.

(c) Show a $k/2$ approximation algorithm. Make sure that your algorithm upper bounds the LP integrality gap.

(d) Show an $\Omega(k)$ lower bound on the integrality gap of your LP.

Problem 3: Don't open too many facilities!

We now consider a variant of the facility location problem, called *k-Median problem*. We are given a metric d on $F \cup C$ and an integer k. Our goal is to open exactly k facilities in F, i.e. choose $F' \subseteq F : |F'| = k$, such that the total cost $\sum_{j \in C} d(F', j)$ is minimized. Now we do not have opening cost for facilities, but we have a strict requirement that we will not open more than k of them.

(a) Assume that d is a line metric. That is, there is a coordinate $\phi(i) \in \mathbb{R}$ for each $i \in F \cup C$ such that $d(i, j) = |\phi(i) - \phi(j)|$. Show a polynomial-time algorithm that computes an optimal solution for this problem (hint: Dynamic programming).

(b) Now we investigate a special case of the problem where the metric d is defined by an unweighted bipartite graph $H = (F \cup C, E)$ such that F and C contain the vertices on the left-hand-side and right-hand-side of the graph respectively. The metric $d(i, j)$ simply evaluates the shortest path distance from i to j in H. First argue that d satisfies three metric properties.

(c) Let us consider an even easier special case when there is an optimal solution having a nice property called *perfect covering property*. This property ensures that the optimal solution costs exactly $|C|$, i.e. there is a collection of facilities $F' = \{f_1, \ldots, f_k\} \subseteq F$ such that $d(F', j) = 1$ for all j. The LP relaxation can be written as follows:

(LP)

$$
\begin{aligned}
\min \quad & \sum_{i \in F} \sum_{j \in C} d(i, j) x_{ij} \\
& x_{ij} \leq y_i \text{ for all } i \in F, j \in C \\
& \sum_{i \in F} y_i = k \\
& \sum_{i \in N_H(j)} x_{ij} = 1 \text{ for all } j \in C
\end{aligned}
$$

Where $N_H(j) \subseteq F$ contains neighboring vertices of j in H. Argue that this LP is a relaxation of the bipartite k-median problem with perfect covering property.

(d) Show a factor of 3 approximation by rounding (LP) for the bipartite k-Median problem with perfect covering property.

(e) *The same as the previous problem, but show a randomized $(1 + 2/e)$ approximation (in expectation) by LP rounding.

(f) Show an integrality gap lower bound of $1 + 2/e - \epsilon$ for arbitarily small $\epsilon > 0$ for bipartite k-Median problem with perfect covering property.

(g) Now back to the bipartite k-Median problem (without perfect covering property). Getting a constant approximation algorithm is not simple here, but if we are allowed to open more than k facilities, it becomes much simpler. For any $\epsilon > 0$, show an $O(1/\epsilon)$ approximation algorithm that opens $(1+\epsilon)k$ facilities. (Hint: see the LP rounding algorithm in the solution of HW1)

Problem 4: Region Growing and Random Cuts

Let $G = (V, E)$ be a complete graph with weight function $w(e)$ and distance function $d(e)$. Let $\alpha, \Delta > 0$ be two parameters. We say that a partition $\pi = \{V_1, \ldots, V_\ell\}$ of vertices is a *α-cheap Δ-small partition* if

$$\sum_{i,j:i<j} \sum_{uv:u \in V_i, v \in V_j} w(uv) \leq \alpha \sum_{uv \in E} d(u,v) w(uv)$$

And diameter of $G[V_i]$ is at most Δ, i.e. for each pair $u, v \in V_i$, we must have $d_G(u,v) \leq \Delta$. In other words, we want to find a partition such that each cluster has small radius, while we do not cut too many edges across the partition (compared to edges in the same set of the partition).

- Use the region growing technique to show how to compute $O(\frac{\log n}{\Delta})$-cheap Δ-small partition for any $\Delta > 0$.
- Use the random cut argument to prove the same thing (with certain probability of success).

Problem 5: Hitting Set, but don't hit me too often!

In the *hitting set problem*, we are given a ground set E and a collection of sets $S_1, \ldots, S_m \subseteq E$. Our goal is to choose a collection of elements $F \subseteq E$ such that for any i, $S_i \cap F \neq \emptyset$, while minimizing the size $|F|$. In other words, we want to hit all the sets using minimum number of elements.

(a) Argue formally that it is the same as set cover. What approximation results follow?

(b) Now consider a variant of hitting set where each set S_i is said to be *satisfied* by $F \subseteq E$ if $|F \cap S_i| = 1$. Our goal is to choose $F : F \subseteq E$ that maximizes the number of satisfied sets (Notice that this is a maximization problem.) For obvious reason, we call this problem *unique hitting set*. Show a constant factor approximation algorithm for unique hitting set when all sets S_i have the same size. Hint: Randomized algorithm.

(c) Use the results from the previous question to show a logarithmic factor approximation algorithm for unique hitting set.

(d) We say that the instance $(E, \{S_i\}_{i=1}^m)$ satisfies the *perfect hiting property* if there is a collection $F \subseteq E$ such that every set is satisfied by F. Given an instance of unique hitting set with perfect hitting property, show an $e/(e-1)$ approximation algorithm by LP rounding.

(e) Suppose that all sets have size 2. Do you think this problem is NP-hard, or there is a polynomial time algorithm? Why? Does it look similar to any problem you learned in class? What if we know that the sets S_i satisfy both $(\forall i)|S_i| = 2$ and perfect hitting property? Is it polynomial time solvable?

(f) Consider a "geometric" version of unique hitting set when the set E contains points on the line, i.e. $E \subseteq \mathbb{R}$, and the set S_i is an interval of the form $[l_i, r_i]$. The question becomes very much of a geometric nature: Choose a subset of points $F \subseteq E$ such that the number of intervals containing exactly one point is maximized. What is your intuition about this problem? Polynomial-time solvable or NP-hard? Give the best approximation algorithm you can think of. Do NOT assume perfect hitting property.