CS-E3190 Principles of Algorithmic Techniques

07. Linear Programming – Graded Exercise

Please read the following **rules** very carefully.

- Do not consciously search for the solution on the internet.
- You are allowed to discuss the problems with your classmates but you should **write the solutions yourself**.
- Be aware that **if plagiarism is suspected**, you could be asked to have an interview with teaching staff.
- The teaching staff can assist with understanding the problem statements, but will **not be giving any hints** on how to solve the exercises.
- In order to ease grading, we want the solution of each problem and subproblem to start on a **new page**. If this requirement is not met, **points will be deduced**.

This problem set revisits the **set cover** problem. You will study a primal-dual algorithm and an LP-rounding algorithm. The last problem connects set cover with vertex cover.

Consider the **weighed set cover** problem. Inputs are a ground set (universe) of n elements $U = \{e_1, e_2, \dots, e_n\}$ and a non-empty family $\mathcal{F} = \{S_1, S_2, \dots, S_m\}$ of m subsets $S_j \subseteq U$. The sets have positive weights $(w_1, w_2, \dots, w_m)$. The goal is to find a *cover* $I \subset \{1, 2, \dots, m\}$ (i.e. I s.t. $\cup_{j \in I} S_j = U$) of minimum cost $c(I) = \sum_{j \in I} w_j$. We assume each instance has a cover[1].

Note that a cover is usually defined as a subfamily $C \subseteq \mathcal{F}$ of sets whose union is U. Here, we consider the set of indices of a cover.

1. **A Primal-Dual set cover algorithm**. Consider the linear programming formulation of the weighted set cover problem. Originally the variables $x_j \in \{0, 1\}$ indicate whether we select set S_j or not. However in the relaxed program (primal) fractional $x_j \in [0, 1]$ variables are allowed. The primal and the corresponding dual are given below.

Primal

$$\begin{aligned} \min_x \quad & \sum_{j=1}^{m} w_j x_j \\ \text{s.t.} \quad & \sum_{j : e_i \in S_j} x_j \geq 1, \quad i = 1, \dots, n \\ & x_i \geq 0, \quad j = 1, \dots, m \end{aligned}$$

Dual

$$\begin{aligned} \max_y \quad & \sum_{i=1}^{n} y_i \\ \text{s.t} \quad & \sum_{i : e_i \in S_j} y_i \leq w_j, \quad j = 1, \dots, m \\ & y_i \geq 0, \quad i = 1, \dots, n \end{aligned}$$

Notice that in the primal we sum over the indices of all sets containing a given element e_i, that is $\{j : S_j \ni e_i\}$. Analogously in the dual we sum over the indices of all elements e_i contained in a given set S_j, that is $\{i : e_i \in S_j\}$.

(a) Derive the dual from the primal and verify that it matches the given dual.
Hint: You can do as in the lectures and collect matrix A, and vectors b and c, or as in the tutorial exercise. Just show the math; no lengthy explanations needed.)

[1] Both algorithms can be extended to report if no cover exits in a straightforward manner. We make this assumption in order to simplify the algorithm descriptions and focus on the linear programming techniques.

(b) Consider the following primal-dual algorithm for set cover.

Algorithm 1: Primal-dual set cover algorithm

input : $\left(U, \mathcal{F}, (w_j)_{j=1}^m\right)$
$y \leftarrow 0$;
$I \leftarrow \emptyset$;
while *some element* e_i *is not covered by* $U_{j\in I} S_j$ **do**
    increase y_i until some dual constraint k, $\sum_{i:e_i\in S_k} y_i = w_k$, is tight;
    update $I = I \cup \{k\}$;
end
output: Selection $I \subset \{1, \dots, m\}$

The algorithm allows "increases" of y_i by zero. Moreover, if multiple constraints become tight simultaneously, one can arbitrarily choose one to be added to I. It should be clear that the algorithm returns a valid cover, given that we assume one exists. The main task of the problem is to establish a bound on the cost $c(I)$.

To this end, let I be the returned cover and y the associated dual variables. Define the *frequency* of an instance as the largest number of sets in $\mathcal{F}$ that cover any one element in U, that is $f = \max_i |\{j : i \in S_j\}|$.

Prove that the cost $c(I)$ is bounded by f times the dual objective:

$$c(I) \leq f \sum_{i=1}^{n} y_i.$$

Hint: For $j \in I$, what connection between w_j and y do we know of?

(c) Let OPT be the minimum cost of a cover. Prove that $c(I) \leq f \cdot OPT$.

2. **An LP-rounding algorithm for set cover**. Consider the same Primal and dual as in part 1, and the following associated algorithm.

Algorithm 2: LP-rounding set cover algorithm

input : $\left(U, \mathcal{F}, (w_j)_{j=1}^m\right)$
construct and solve the **primal** linear program;
$x^* \leftarrow$ the optimal *fractional* solution to the primal;
$f \leftarrow \max_i |\{j : i \in S_j\}|$;
$I \leftarrow \emptyset$ initialize empty cover;
for *j = 1, ..., m* **do**
    if $x_j^* \geq 1/f$ **then**
        $I \leftarrow I \cup \{j\}$;
end
output: Set $I \subset \{1, \dots, m\}$

(a) Prove the algorithm is correct, i.e. that it returns a *cover* I.
Hint: What do we know about x^*?

(b) Prove that $c(I) \leq f \cdot OPT$ where OPT is the minimum cost of a cover.