CSC373— Algorithm Design, Analysis, and Complexity — Spring 2018

Solutions for Tutorial Exercise 12: Linear Programming and Approximation Algorithms

---

LP and Dual LP for Weighted Vertex Cover. Suppose we are given a graph $G = (V, E)$ and a set of non-negative, integer-valued weights $\{w(v) \mid v \in V\}$, where each $w(v)$ denotes the weight of vertex v. We seek a vertex cover $S \subseteq V$ such that the weight of S, i.e.,

$$w(S) = \sum_{v \in S} w(v), \tag{1}$$

is as small as possible. For a given graph, we denote a minimum weight vertex cover as S^* and its value as $W^* = w(S^*)$.

a) Show that this minimization problem can be expressed as an integer linear programming (ILP) problem. Moreover, write this ILP in the following form (i.e., we are anticipating this problem is the dual of some other maximization LP problem written in standard form, see part (b) below):

$$\text{minimize } \vec{b}^T \vec{y} \tag{2}$$

subject to,

$$\vec{y}^T A \geq \vec{c}^T, \tag{3}$$

$$\vec{y} \geq \vec{0}. \tag{4}$$

For each vertex $v \in V$, there should be one LP variable, say $y(v)$, and this variable is associated with the v^{th}-row of A. (Or, if you prefer, you can number the vertices from 1 to $|V|$, and number the edges from 1 to $|E|$, and translate the indices v and e below to the corresponding integer indices.) If $y(v) = 1$ this represents the case in which v is selected for the vertex cover, and otherwise $y(v) = 0$. Also, each column of A is associated with an edge $e = (u, v) \in E$. (You do not need to impose the constraint that $y(v) \leq 1$, as this should be a consequence of the minimization and other constraints.) Explain what the appropriate values of A, b and c are.

Solution. Let $V = \{v_k\}_{k=1}^n$, $E = \{e_k\}_{k=1}^m$. Let $y_k \in \{0, 1\}$, where $y_k = 1$ iff v_k is an element of the vertex cover S. Then $b = (w(v_1), \ldots, w(v_n))^T$, c is the n-vector of all ones, and

$$A_{i,j} = \begin{cases} 1, & \text{if } e_j = (v_k, v_i) \text{ or } e_j = (v_i, v_k), \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

The j^{th} column of $y^T A \geq 1^T$ imposes the constraint on the edge $e_j = (v_a, v_b)$ that, $y_a + y_b \geq 1$, that is, at least one of y_a or y_b must be 1.

A more compact notation is to use the objects v and e as indices themselves. In this case the elements of y are $y(v)$, and the elements of A are $A_{v,e} = \delta(v \in e)$. So, for example, the e^{th} column of $y^T A$ is

$$y^T A_{*,e} = \sum_{v \in V} y(v) A_{v,e} = \sum_{v \in V} \delta(v \in e) = y_a + y_b, \quad \text{where } e = (a, b).$$

b) Write out the primal linear program associated with the LP in part (a). The elements of the unknown vector p associated with the columns of A and, as a result, are associated with the edges in the graph. We refer to each of these entries as $p(e)$.

Define E_v to be the set of edges in E that end at v. Show that the feasibility constraints of the primal problem are equivalent to

$$\sum_{e \in E_v} p(e) \leq w(v), \text{ for all } v \in V, \text{ and} \tag{6}$$

$$p(e) \geq 0, \text{ for all } e \in E. \tag{7}$$

Solution. The primal problem associated with part (a) is in standard form, minimize $c^T p$ subject to $Ap \leq b$ and $p \geq 0$.

From part (a) we have $b_i = w_i$. Also, from eqn (5), the i^{th} row of A has the value 1 for every edge e_j that has v_i as an endpoint, and is 0 otherwise. That is, the i^{th} row of A is 1 for $e_j \in E_{v_i}$. Therefore 6 follows.

c) Explain how an integer-valued feasible solution to the problem in part (b) provides a lower bound on the minimum weight W^* for any vertex cover of the given graph. Specifically, show that

$$\sum_{e \in E} p(e) \leq W^*. \tag{8}$$

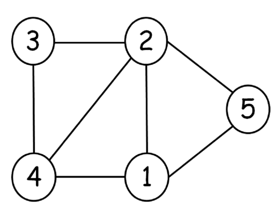
Solution. Note the Mutual Bound Theorem of Linear Programming (slide 5 of the lecture notes on Duality in Linear Programming) applies to any feasible solutions of the primal and dual problems, including integer-valued solutions.

Equation (8) follows directly from the Mutual Bound Theorem, since $\sum_{e \in E} p(e)$ is a feasible profit for the primal problem in part (b), and W^* is a feasible profit for the associated dual linear program problem in part (a).

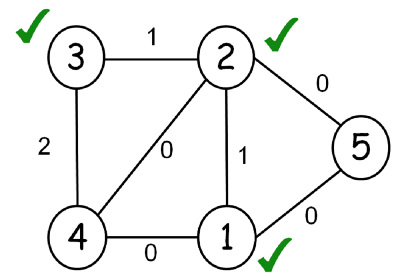
d) Consider the example graph shown below, where the weights $w(v)$ are displayed in each vertex. We define a "maximal" solution to the the primal problem in part (b) to be a feasible solution such that each $p(e)$ is individually increased as much as possible, assuming the other p's are all held fixed. Note this is not necessarily a solution which maximizes the profit of the LP. Rather, it is just a feasible solution for which simple, individual increases of the variables $p(e)$ cannot achieve a larger solution.

Find an integer-valued, maximal, feasible solution to the primal problem in part (b). You can find such a solution simply by selecting an ordering for all the edges in E, loop through these edges, and select the maximum possible value of $p(e)$ for each edge individually. The result will be a maxim**al** feasible solution (but not a maximum soluton).

Represent your solution simply by redrawing the graph below and writing the value $p(e)$ beside each edge. On this drawing indicate the vertices at which the feasibility constraints (6) are equality constraints, along with the edges for which (7) are equality constraints. Perhaps do this simply by placing check marks near these vertices and edge weights in your drawing. We refer to these vertices and edges as "tight" (in the primal problem).



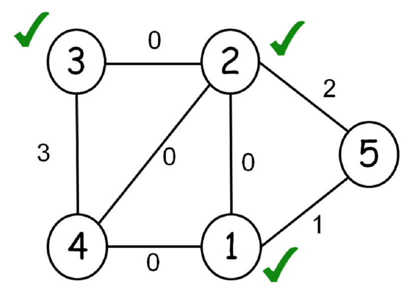
Solution. Suppose we consider the edges in the order $(1, 2)$, $(1, 4)$, $(1, 5)$, $(2, 3)$, and so on. (Here we are abusing the notation and referring to vertices by their weights, this works in the above example because all the weights are distinct. Also, the above ordering of the edges is then simply a lexicographic ordering of the edges in the graph in terms of the weights of their endpoints. We choose this ordering for this illustration, any ordering will do.) For each edge in the list, we maximize $p(e)$ holding the values $p(e')$ constant, for all other edges $e' \in E\ \{e\}$. This gives the result shown below. Note that a different result could be produced if we considered the edges in a different order. However, the result is always a feasible solution to the primal problem.



The profit for the primal problem for the above solution is $\sum_{e \in E} p(e) = 4$. (Note that, by (8), the minimum weight vertex cover must have weight ≥ 4.)

e) Try modifying your maximal solution in part (d) to reduce the number of edges $e \in E$ that: 1) connect two tight vertices; and 2) have $p(e) > 0$. One way to modify a solution is to pick a vertex $v \in V$, and "rotate" the weights on two edges that end at v. Such a rotation consists of adding an integer-valued constant to $p(e)$ on one edge that terminates at v, and subtracting this same constant from another edge that terminates at v, maintaining the feasibility of the resulting solution. After doing a few of these rotations it is sometimes possible to increase individual values of $p(e)$, and recover a new maximal feasible solution. Try this procedure on your example from part (d) and draw the resulting graph, indicating the new values of $p(e)$, and which vertices are tight.

Solution: Another feasible solution is below. For this example it is possible to satisfy the constraint that no edge with $p(e) > 0$ has two saturated endpoints.



The primal profit for this solution is $\sum_{e \in E} p(e) = 6$. (Note this now to agrees with the minimum possible weight $w(S^*)$ of a vertex cover for this problem.)

f) For your maximal feasible solution in part (d) or (e), define S to be the set of vertices $v \in V$ which are tight, that is,

$$\sum_{e \in E_v} p(e) = w(v). \tag{9}$$

Show that S is a vertex cover. Do this first for your examples in parts (d) and (e), then try to show this is true in general.

Solution: In the previous examples $S = \{1, 2, 3\}$ (where again, we are abusing the notation, and indicating vertices by their weights). This is a vertex cover.

To see that set S will always be a vertex cover, suppose that it isn't. That is, there exists an edge $e = (v_a, v_b)$ such that neither vertex v_a or v_b is in S. But this contradicts the definition of the algorithm which maximizing $p(e)$ on each edge, and never decreases any of the values $p(e)$.

g) Define $w(S) = \sum_{v \in S} w(v)$, where S is as in part (f). Show

$$w(S) = \sum_{v \in S} \sum_{e \in E_v} p(e). \tag{10}$$

Solution: The vertices $v \in S$ are sharp, this means $\sum_{e \in E_v} p(e) = w(v)$, and equation (10) follows.

h) For each edge $e \in E$, let $N(e)$ be the number of times the value $p(e)$ is counted on the right hand side of (10). Show

$$w(S) = \sum_{e \in E} N(e)p(e). \tag{11}$$

What's the maximum of $N(e)$ for all e? Note that equations (8) and (11) were the motivation for part (e), where we tried to minimize the number of edges e with $p(e) > 0$ and for which both endpoints were "tight".

Solution: The result follows from (10) with $N(e)$ equals the number of times the edge e ends at a sharp vertex $v \in S$. So $N(e)$ can be 1 or 2, (but not 0 since each edge ends in at least one sharp vertex).

i) Show $w(S) \leq 2W^*$, where W^* is the minimum weight of any vertex cover. We can therefore conclude:

$$B = \sum_{e \in E} p(e) \leq W^* \leq W(S) = \sum_{v \in S} w(v) = \sum_{e \in E} N(e)p(e) \leq 2W^*, \tag{12}$$

and it follows that this approach gives a 2-approximation for weighted Vertex Cover.

Solution: Since $N(e) \leq 2$, we have $\sum_{e \in E} N(e)p(e) \leq 2 \sum_{e \in E} p(e) = 2B$. The result $2B \leq 2W^*$ follows from (8).

j) In the lecture notes on Linear Programming we described complementary slackness for constraints on optimal vertices of the primal and dual problems. We do not typically have optimal vertices here, for one because these optimal solutions could be non integer-valued and also because we did not explicitly try to find optimal solutions to either ILP problem (why not?). However, we can still look at the formalism of complementary slackness. That is, given the tight constraints in the primal problem in part (b), which constraints of the dual problem in part (a) would complementary slackness then require to be tight (and which can be loose)? How does this compare to the above choice of S?

Solution: The tight constraints of the primal problem are exactly the rows for which equation (6) or (7) are equalities. These rows are indicated by specific tight vertices or tight edges, respectively.

Complementary slackness for this example indicates that the tight vertices for the primal problem are not tight in the dual problem. That is, a vertex is tight in the primal problem iff it is loose in the dual problem, and similarly for edges. Note that a vertex being loose in the dual problem (in part (a)) means that the sign constraint $y(v) = 0$ need not apply. This corresponds to the choice made by the algorithm to set $y(v) = 1$ for those vertices.

k) Briefly compare this approach to the "Pricing Method for Vertex Cover", given at the end of the lecture notes on Approximation Algorithms.

Solution: It is the same method.

Note that this approach gives more information about the optimum value W^* than simply the 2-approximation $w(S)$. Specifically, from equation (12) we see this approach also gives a poly-time computable lower bound B on W^*. It follows from (12) that this ratio of $w(S)$ and the lower bound B is always[1] less than or equal to 2. This ratio can often be significantly less that 2, and can sometimes be one (which would certify that S is actually an optimal solution). Moreover, $w(S)/B$ serves as a poly-time computable upper bound for the approximation ratio for the given input problem and, as we said, this ratio $w(S)/B$ is always[1] less than or equal to 2.

[1] Except in trivial examples for which the maximal solution $p(e)$ is identically zero.