Linear Programming 6

Ted Tinker, 3223468

May 28, 2017

Problem 1

Part A

We would like to minimize the cardinality of a set of vertices C, so let $y_v = 1$ if vertex $v \in V$ is in C, and $y_v = 0$ if it is not. Then, the objective function is

$$minimize \sum_{v \in V} y_v.$$

To ensure each edge $e \in E$ has at least one point in C (with $e = vw$) :

$$y_v + y_w \geq 1, \text{ so either } v, w, \text{ or both} \in C.$$

To make an integer program, we allow $y_v \geq 0$, $y_v \in \mathbb{Z}$. As we are minimizing the sum of the y_v, necessarily they will be in the set $\{0, 1\}$, such that each point is either in C or not.

Part B

To solve this, let's find the primal LP of the 4-vertex case in matrix form: $minimize \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$ so

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \geq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Its dual is: $maximize \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} x_{y_1,y_2} \\ x_{y_1,y_3} \\ x_{y_1,y_4} \\ x_{y_2,y_3} \\ x_{y_2,y_4} \\ x_{y_3,y_4} \end{bmatrix}$ subject to

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} x_{y_1,y_2} \\ x_{y_1,y_3} \\ x_{y_1,y_4} \\ x_{y_2,y_3} \\ x_{y_2,y_4} \\ x_{y_3,y_4} \end{bmatrix} \geq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Each $x_{v,w}$ is an edge from v to w. In a relaxation, we will only specify that $x_{v,w}$ is non-negative.

In the general dual, the objective function becomes $maximize \sum_{e \in E} x_e$. The constraints will demand

$$\sum_{e \in \delta(y_v)} x_e \geq 1, \text{ for } v \in V.$$

This confirms that each vertex is connected to at least one edge $e = v, w$ so either v, w, or both are in C. Declare only $x_e \geq 0$.

Problem 2

Part A

The edges in this graph are $\{s, a\}$, $\{s, b\}$, $\{a, b\}$, $\{a, c\}$, $\{b, d\}$, $\{c, d\}$, $\{c, t\}$, and $\{d, t\}$. Let $x_0 = \{0, 0, 0, 0, 0, 0, 0, 0\}$, respectively, and $U_0 = \{s\}$. $slack_{x_0}(sa) = 3 - 0 = 3$, $slack_{x_0}(sb) = 4 - 0 = 4$. Choosing the edge with minimum slack, sa, gives us the next iteration:

$x_1 = (3, 0, 0, 0, 0, 0, 0, 0)$, $U_1 = \{s, a\}$. $slack_{x_1}(sb) = 4 - 3 = 1$, $slack_{x_1}(ab) = 2 - 3 = -1$, $slack_{x_1}(ac) = 6 - 3 = 3$. Again choosing the edge with minimum slack, ab, the next iteration is:

$x_2 = (3, 0, 2, 0, 0, 0, 0, 0)$, $U_2 = \{s, a, b\}$. Now, $slack_{x_2}(sb) = 4 - 0 = 4$—and I'm not even sure we need to check this one, as s and b are already in U_2. $slack_{x_2}(ac) = 6 - 3 - 2 = 1$. $slack_{x_2}(bd) = 2 - 2 = 0$. Next iteration:

$x_3 = (3, 0, 2, 0, 2, 0, 0, 0)$, $U_3 = \{s, a, b, d\}$. Now, $slack_{x_3}(ac) = 6 - 3 - 2 = 1$, $slack_{x_3}(dc) = 1 - 2 = -1$. $slack_{x_3}(dt) = 7 - 2 = 5$. Next iteration:

$x_4 = (3, 0, 2, 0, 2, 1, 0, 0)$, $U_4 = \{s, a, b, c, d\}$. Now, $slack_{x_4}(ct) = 3 - 1 = 2$, while $slack_{x_4}(dt) = 7 - 2 - 1 = 4$. Next iteration:

$x_5 = (3, 0, 2, 0, 2, 1, 3, 0)$, $U_5 = \{s, a, b, c, d, t\}$. This contains s and t, so it necessarily contains an st path. By choosing the correct vertices, we find an st path of minimum cost: sb, bd, dc, ct, with a cost of 10.

Part B

$x_0 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$, representing edges sa, sb, sc, sd, ab, ac, ae, bd, bt, cd, cf, dg,, ef, et, fg, gt. Let $U_0 = \{s\}$, because our path starts at s. The first slacks to check are $slack_{x_0}(sa) = 4$, $slack_{x_0}(sb) = 6$, $slack_{x_0}(sc) = 3$, and—the minimum—$slack_{x_0}(sd) = 1$.

Assign $x_1 = (0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$, $U_1 = \{s, d\}$. $slack_{x_0}(sa) = 4 - 1 = 3$, $slack_{x_0}(sb) = 6 - 1 = 5$, $slack_{x_0}(sc) = 3 - 1 = 2$, $slack_{x_1}(db) = 5 - 1 = 4$, $slack_{x_1}(dc) = 4 - 1 = 3$, and $slack_{x_1}(dg) = 7 - 1 = 6$. The next iteration is

$x_2 = (0, 0, 3, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$, $U_2 = \{s, d, c\}$. $slack_{x_2}(sa) = 4 - 1 - 3 = 0$, $slack_{x_2}(sb) = 6 - 1 - 3 = 2$, $slack_{x_2}(ca) = 2 - 3 = -1$, $slack_{x_2}(cf) = 2 - 3 = -1$, $slack_{x_2}(db) = 5 - 1 = 4$, and $slack_{x_2}(dg) = 7 - 1 = 6$. The next iteration is

$x_3 = (0, 0, 3, 1, 0, 2, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0)$, $U_3 = \{s, d, c, f, a\}$. (Looking ahead in the graph, I can see moving to f on the way to t with be cheaper than moving to a first, but in this step, I include both of them in U.) $slack_{x_3}(ab) = 3 - 2 = 1$, $slack_{x_3}(ae) = 4 - 2 = 2$, $slack_{x_3}(sb) = 6 - 3 - 1 = 2$, $slack_{x_3}(dg) = 7 - 1 = 6$, $slack_{x_3}(fe) = 2 - 2 = 0$, and $slack_{x_3}(fg) = 3 - 2 = 1$. The next iteration is

$x_4 = (0, 0, 3, 1, 0, 2, 0, 0, 0, 0, 2, 0, 2, 0, 0, 0)$, $U_4 = \{s, d, c, f, a, e\}$. From here, we obviously take the 1-cost edge et. This gives the minimum-cost st-path, sc, cf, fe, et, costing 8.