Problem Set 5 Solutions

Wed, October 21, 2009

Problem 1. For reference, the linear program is as follows:

$$\begin{aligned} \text{minimize } & cx \\ \text{s.t. } & x_1 + x_2 \geq 1 \\ & x_1 + 2x_2 \leq 3 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \\ & x_3 \geq 0 \end{aligned}$$

(a) For $c = (-1, 0, 0)$, this minimizes the quantity $-x_1$, so it maximizes x_1. We have $x_1 + 2x_2 \leq 3$ with $x_2 \geq 0$, so the most x_1 can be is 3. This sets x_2 to 0, and all the other constraints are satisfied if $x_3 \geq 0$. Thus we have the optimal value of -3 and the set of possible solutions $\{(3, 0, x_3) : x_3 \geq 0\}$.

(b) For $c = (0, 1, 0)$, this minimizes the quantity x_2. Given that $x_2 \geq 0$, we set x_2 to 0. The rest of the constraints force $1 \leq x_1 \leq 3$ and $x_3 \geq 0$. Thus we have the optimal value of 0 and the set of possible solutions $\{(x_1, 0, x_3) : 1 \leq x_1 \leq 3, x_3 \geq 0\}$.

(c) For $c = (0, 0, -1)$, this minimizes the quantity x_3, so it maximizes x_3. Since the only constraint on x_3 is that it be nonnegative, the solution is unbounded and has an optimal value of $-\infty$. The set of possible solutions is $\{(x_1, x_2, \infty) : x_1 + x_2 \geq 1, x_1 + 2x_2 \leq 3, x_1 \geq 0, x_2 \geq 0\}$.

Problem 2. **(a)** Let x_i be the variable that denotes the amount of money we wish to trade with client i. We want to maximize the amount of yen we make, so we want to maximize the sum of the x_i's that trade from anything to yen, which we denote by y, minus the amount of yen we trade away. Making sure we optimize for the final amount of yen is important because otherwise we would simply optimize for a maximum amount of yen we could achieve over time, which could mean trading away all the yen we got into pesos and trading it back into yen again, increasing the LP's optimal value but not actually getting more yen. We denote by d the dollar currency. When we write $i = (a_i, b_i)$, we mean the client i that trades currency a_i for b_i. We can write that as:

$$\max \left(\sum_{i=(a_i,y)} r_i x_i - \sum_{i=(y,a_i)} y \right)$$

We subject the variables x_i to a few constraints. First, we have that we cannot trade more than x_i dollars with client i:

$$0 \leq x_i \leq u_i$$

Lastly, we cannot trade more of a currency than we have for every currency c:

$$\sum_{i=(a_i,c)} r_i x_i \geq \sum_{i=(a,b_i)} x_i$$

Also, we cannot trade more than D dollars, as that is our initial amount of money, plus whatever we trade into dollars:

$$\sum_{i=(d,a_i)} x_i \leq D + \sum_{i=(a_i,d)} r_i x_i$$

(b) We can consider the solution to the LP as a flow on a graph with currencies as nodes and clients as edges. The flow conservation rules are not the same as that of max-flow due to the multiplicative factor.

We can decompose the flow into paths. Given any path in the graph starting from s, we compute the currency that is effectively traded by the path. This computation involves finding the minimum of $f(e_i)/\prod_{j<i} r(e_j)$. We will end up with a path and cycle decomposition. The paths can be executed one after another till all operations are performed. Cycles do not earn money due to the "no profit by arbitrage" condition, so we can ignore them. Thus no borrowing is necessary.

(c) Paths that do not end in yen can be ignored. Thus we can construct a flow based on the paths we extracted. This simplified flow will have the same value and will end the day with only dollars and yen. Alternatively, we can modify our linear program to constrain that for all types of currencies not dollars or yen, we have to trade away exactly the amount that we get from other curriences. This is still feasible (we just don't trade the extra dollars that would have become the useless amounts of currency at the end), and it is still optimal because those amounts of currency do not affect the optimum value of the LP.

Problem 3. Let $E = \{(i,j) \mid \text{a packet needs to be routed from } i \text{ to } j\}$. Here, $m = |E|$.

(a) Observe that the $2n$ equality constraints are all tight, but only $2n-1$ of them are independent (because the sum of all the row sums equals the sum of all the column sums). But we are working in an m-dimensional space, so $m-2n+1$ of the inequality constraints must be tight. In other words, $m-2n+1$ of the x_{ij} must equal zero.

(b) If $(i,j) \notin E$, then $M_{ij} = 0$. This leaves m potentially nonzero entries. If at least two of these are nonzero on every row, then the total number of zero entries is at most $n^2 - 2n$, of which we have already accounted for $n^2 - m$. This leaves at most $m - 2n$ zero entries $x_{ij} = M_{ij}$ such that $ij \in E$.

But by (a), there are at least $m-2n+1$ of the x_{ij} are zero. This is a contradiction, so some row a must have at most one nonzero entry.

The entire row cannot be zero (because of the row sum property), so there must be exactly one nonzero entry $x_{ab} = 1$. Because the entries of the matrix are nonnegative, all the other entries in column b must be zero as well.

(c) Consider the matrix M' obtained by deleting row a and column b. The sum of each row and column of this matrix is still 1, because we have only deleted a zero entry. Therefore, M' is doubly stochastic.

If R_a and C_b represent the positions in row a and column b respectively, then let $p = |E \cap (R_a \cup C_b)|$. In going from M to M', we lost p variables. We also lost 2 equality constraints (for the row and the column) and $p-1$ tight inequality constraints. However, these $p+1$ constraints are not linearly independent, because if we set all but one entry in a row or a column, the remaining entry is determined by the sum constraint. Therefore, we have taken away at most p linearly independent constraints, so that M' is a vertex for the resulting LP.

By the induction hypothesis, M' has integer entries, so that M has integer entries.

For the base case, note that for $n = 1$, the only doubly stochastic matrix is (1), which satisfies the given condition. So our claim is true by induction.

A one-zero matrix corresponds to a matching between row i and column j whenever $x_{ij} = 1$. Because any feasible point of the demand polytope can be written as a convex combination of vertices, our switch can always decompose any demand into a convex combination of matchings, in which the sum of the weights of the individual matchings is $\leq \lambda$. So as long as it can deliver matchings at rate λ, it can deliver the specified traffic.

Problem 4. Let the variables be $\mathbf{x} = (x_1, \cdots, x_n)^T, \mathbf{x}' = (x'_1, \cdots, x'_{n'})^T$ and $\mathbf{x}'' = (x''_1, \cdots, x''_{n''})^T$, where $\mathbf{x}' \geq \mathbf{0}$, $\mathbf{x}'' \leq \mathbf{0}$, and $\mathbf{x}$ is unrestricted in sign. Write the constraints as

$$\begin{aligned} A_{11}\mathbf{x} + A_{12}\mathbf{x}' + A_{13}\mathbf{x}'' &= \mathbf{b} \\ A_{21}\mathbf{x} + A_{22}\mathbf{x}' + A_{23}\mathbf{x}'' &\leq \mathbf{b}' \\ A_{31}\mathbf{x} + A_{32}\mathbf{x}' + A_{33}\mathbf{x}'' &\geq \mathbf{b}'' \end{aligned}$$

An arbitrary optimization LP calls for the minimum or maximum value of a linear objective function. Without loss of generality, consider a minimization problem (a maximization problem can be converted into a minimization problem by switching the signs of the coefficients on the objective function). Write the objective function as

$$\mathbf{c} \cdot \mathbf{x} + \mathbf{c}' \cdot \mathbf{x}' + \mathbf{c}'' \cdot \mathbf{x}''.$$

Define dual variables $\mathbf{y}$, $\mathbf{y}'$, and $\mathbf{y}''$ of appropriate sizes where $\mathbf{y}$ is unrestricted in sign, $\mathbf{y}' \leq \mathbf{0}$, and $\mathbf{y}'' \geq 0$. The dual constraints are

$$\begin{array}{rcl} \mathbf{y}^T A_{11} + \mathbf{y}'^T A_{21} + \mathbf{y}''^T A_{31} & = & \mathbf{c}^T \\ \mathbf{y}^T A_{12} + \mathbf{y}'^T A_{22} + \mathbf{y}''^T A_{32} & \leq & \mathbf{c}'^T \\ \mathbf{y}^T A_{13} + \mathbf{y}'^T A_{23} + \mathbf{y}''^T A_{33} & \geq & \mathbf{c}''^T. \end{array}$$

Multiplying the primal constraints by the dual variables, we get

$$\begin{array}{rcl} \mathbf{y}^T(A_{11}\mathbf{x} + A_{12}\mathbf{x}' + A_{13}\mathbf{x}'') & = & \mathbf{y} \cdot \mathbf{b} \\ \mathbf{y}'^T(A_{21}\mathbf{x} + A_{22}\mathbf{x}' + A_{23}\mathbf{x}'') & \geq & \mathbf{y}' \cdot \mathbf{b}' \\ \mathbf{y}''^T(A_{31}\mathbf{x} + A_{32}\mathbf{x}' + A_{33}\mathbf{x}'') & \geq & \mathbf{y}'' \cdot \mathbf{b}''. \end{array}$$

These three (in)equations sum to

$$\begin{pmatrix} \mathbf{b} \\ \mathbf{b}' \\ \mathbf{b}'' \end{pmatrix} \cdot \begin{pmatrix} \mathbf{y} \\ \mathbf{y}' \\ \mathbf{y}'' \end{pmatrix} \leq \begin{pmatrix} \mathbf{y} \\ \mathbf{y}' \\ \mathbf{y}'' \end{pmatrix}^T \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{x}' \\ \mathbf{x}'' \end{pmatrix} \leq \begin{pmatrix} \mathbf{c} \\ \mathbf{c}' \\ \mathbf{c}'' \end{pmatrix} \cdot \begin{pmatrix} \mathbf{x} \\ \mathbf{x}' \\ \mathbf{x}'' \end{pmatrix},$$

which is just the statment of weak duality.

Problem 5. The dual of this program is:

$$\min \sum_{e \in E} u_e x_e \text{ subject to } \left\{ \begin{array}{rcl} \sum_{e \in P} x_e & \geq & 1 \quad \forall\ s-t \text{ paths } P \\ x_e & \geq & 0 \end{array} \right.$$

If we force the variables $x_e \in \{0,1\}$, then we see we have an integer program that computes the minimum cut in the graph. The dual program is just a relaxation of this IP with $x_e \in [0,1]$, since $x_e > 1$ does not occur in an optimal solution. Note that we know that the LP above can achieve the minimum cut value. By strong duality, though, we know that the minimum cut value is the optimum value of this LP, since the primal LP computes the value of the maximum flow in the graph.