Linear Programming and Semidefinite Programming **CMU 15-859E, Fall 2011**

HOMEWORK 3
Due: Tuesday, October 11.

Ground rules: *same as for Homework 1.*

---

0. (Exercises.) Challenge to a Dual. Write down the dual of the following LP:

$$\begin{aligned} \text{maximize} \quad & 10x_1 - 2x_2 + x_3 - 11x_4 \\ \text{subject to} \quad & 2x_1 - 5x_2 - 2x_3 + 9x_4 \leq 12 \\ & 5x_1 - 14x_2 - 5x_3 + x_4 \geq -31 \\ & 3x_1 + 4x_2 + x_3 - x_4 = 9 \\ & x_1, x_2 \geq 0 \\ & x_3 \leq 0 \\ & x_4 \in \mathbb{R} \end{aligned}$$

Write the dual of the resulting dual, and show it is equal to the primal.

0′. Go with the Flow. Consider the max-flow LP(3.1) from Lecture 3.

(a) Show that it is equivalent to the path-based max-flow LP from Lecture 6: on any instance of the max-flow problem, show that a solution to one can be transformed into a solution to the other, having the same objective function value.

(b) If you change the flow conservation equality in LP(3.1) to $\leq$ (total out-flow is *at least* the total in-flow at each vertex), show that the optimal value of the LP does not change.

(c) Write down the dual of LP(3.1). Use the name "$\ell_{(u,v)}$" for the dual variable corresponding to the inequality $f_{(u,v)} \leq C_{(u,v)}$, and use the name "π_v" for the dual variable corresponding to conservation of flow at v.

(d) Give a natural interpretation of the dual variables: what does π_v "mean"? Show that the dual is equivalent to the dual for the path-based dual from Lecture 6, again by mapping solutions of one to those of the other.

0″. An Infeasible Pair. Give an example of a primal-dual pair of LPs that are both infeasible.

1. Reductio ad Solutionem de Feasibility (Redux). In Hwk2(#3), we saw how to reduce the problem of solving a general LP to just the decision version of feasibility. Now we consider an even more general *certification version* of solving the LP $\min\{c^T x \mid Ax \geq b\}$, which we call CERT-SOLVE-LP. It takes as input the $m \times n$ matrix A, the vectors $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$. The desired output now is:

- If the LP is infeasible: output `Infeasible` and a vector $y \in \mathbb{R}^m$ such that $y \geq 0$, $y^\top A = 0$, and $y^\top b > 0$.
- If the optimal value is $-\infty$: output `Unbounded` and two vectors $x, r \in \mathbb{R}^n$ such that $Ax \geq b$ and $Ar \geq 0$, and $c^\top r < 0$. Note that this gives us a ray $\{x+\lambda r \mid \lambda \in \mathbb{R}_{\geq 0}\}$ such that $A(x+\lambda r) \geq b$ for every point along this ray, and the objective function value $c^\top(x+\lambda r) \to -\infty$ as $\lambda \to \infty$.
- If the LP is feasible and the optimum is bounded, output a vector $x \in \mathbb{R}^n$ which is an optimal feasible solution to the LP, and a feasible solution $y \in \mathbb{R}^m$ to the dual such that $c^\top x = b^\top y$.

(a) Give a reduction from Cert-Solve-LP to the problem Solve-LP from Hwk2(#3). Remember the latter only outputs `Infeasible`, `Unbounded`, or the optimal solution. Ideally you should make only constant number of calls to Solve-LP, each time with LPs having only $O(m+n)$ variables and constraints.

(b) Compose the reductions from Hwk2(#3) and the one above to infer that if we can solve the decision version of polyhedron feasibility in polynomial time, we can solve Cert-Solve-LP in polynomial time. Suppose we could solve the decision problem in *strongly* poly-time: does this sequence of reductions give a strongly poly-time algorithm for Cert-Solve-LP? Why or why not?

Hint: how many calls did binary search need?

(c) Recall the strong duality theorem says that if a (primal) LP is feasible and bounded, then its dual LP is also feasible and bounded, and has the same optimal value as the primal.

Show how to take an linear program $\min\{c^\top x \mid Ax \geq b\}$ with m variables and n constraints (call it LP1), and write a new linear program LP2 with $O(m+n)$ variables and constraints, such that LP2 is feasible if and only if LP1 is feasible and bounded. Moreover, if LP2 is feasible, its optimal value should equals LP1's optimal value.

(d) Now use part (c) to give a reduction from Solve-LP (which takes an LP and outputs `Infeasible`, `Unbounded`, or the optimal feasible solution x) to the search version of polyhedron feasibility.

Infer that giving a strongly polytime algorithm for the decision version of polyhedron feasibility implies a strongly polytime algorithm for Cert-Solve-LP.

2. Alternate Forms of Farkas' Lemma. We saw in class (and you proved in Hwk2) that given a system of linear inequalities

$$Ax \leq b, x \geq 0$$

with $A \in \mathbb{R}^{m \times n}$, exactly one of the following two possibilities are true: this system of inequalities is feasible, or there exists $y \in \mathbb{R}^m$ such that $y \geq 0$, $y^\top A \geq 0$, but $y^\top b < 0$. Use this form to prove the following two statements:

(a) Given a system of linear inequalities $Ax \leq b$, either it is feasible, or there exists $y \in \mathbb{R}^m$ such that $y \geq 0$, $y^\top A = 0$, but $y^\top b < 0$.

(b) Given a system of linear inequalities $Ax = b, x \geq 0$, either it is feasible, or there exists $y \in \mathbb{R}^m$ such that $y^\top A \geq 0$, but $y^\top b < 0$.

3. As Simplex as Possible, but no Simpler! We claimed that the simplex algorithm (with the right pivoting rule), given the system $\max\{c^\top x \mid Ax = b, x \geq 0\}$, walks from bfs to bfs, until it terminates at a bfs (corresponding to $B \subseteq [n]$) where the coefficients of all non-basic variables in the last line of the tableau were non-positive. Recall that $N = [n] \setminus B$.

(a) Since we are at a basic solution corresponding to the columns in B, infer that the first m rows of the tableau correspond to

$$x_B = A_B^{-1} b - A_B^{-1} A_N x_N.$$

(b) Infer that the last line of the tableau must read

$$z = c_B^\top (A_B^{-1} b - A_B^{-1} A_N x_N) + c_N^\top x_N.$$

(c) Hence infer that the coefficients of the non-basic variables in the last line of the tableau are given by the entries of the row vector

$$c_N^\top - c_B^\top A_B^{-1} A_N.$$

(d) If all these coefficients are non-positive, show that the solution obtained by setting $x_B = A_B^{-1} b$ and $x_N = 0$ is an optimal solution to the LP. (Hint: prove by contradiction.) Note this solution has value $c_B^\top A_B^{-1} b$.

(e) Now consider the dual LP $\max\{b^\top y \mid A^\top y \geq c\}$. Show that $y = (c_B^\top A_B^{-1})^\top$ is a feasible solution to the dual LP, and has value equal to $c_B^\top A_B^{-1} b$.

This gives another proof of the duality theorem, assuming that the simplex algorithm always terminates on any bounded and feasible LP.

4. To b-Matched, or not to b-Matched? An instance of the bipartite b-matching problem is a bipartite graph $G = (U, V, E)$, where each vertex $i \in U \cup V$ has a demand $b_i \in \mathbb{Z}_{\geq 0}$, such that $\sum_{i \in U} b_i = \sum_{i \in V} b_i = B$. A *$b$-matching* is a function $M : E \to \mathbb{Z}_{\geq 0}$ such that for every $i \in U \cup V$,

$$\sum_{\{i,j\} \in E} M(\{i, j\}) = b_i.$$

Show that there exists a b-matching if and only if

$$\sum_{i \in C} b_i \geq B$$

for every vertex cover $C \subseteq U \cup V$.