ORIE 6300 Mathematical Programming I September 9, 2008

Lecture 4

Lecturer: David P. Williamson *Scribe: Anjie Guo*

Recall the maximum flow problem from last lecture. The maximum flow problem is defined by a directed graph $G = (V, A)$, and two distinguished nodes, s (the *source*) and t (the *sink*). The goal of the problem is to send as much flow as possible from the source to the sink such that each arc carries at most one unit of flow, and for every node of the graph other that the source and sink, the total amount of flow entering the node is equal to the amount leaving the node.

Last time we formulated the maximum flow problem as a linear program where the variables correspond to paths from s to t. For each such path, P, we will have a variable x_P. Let $\mathcal{P}$ denote the set of all paths from s to t. The variable x_P states how much flow is sent from the source to the sink on path P. The constraints express that for each arc $(u, v) \in A$, at most one unit of flow can be sent through arc (u, v):

$$\begin{aligned}
\max \sum_{P \in \mathcal{P}} & x_P \\
x_P & \geq 0 \text{ for each path } P \in \mathcal{P} \\
\sum_{P:(u,v)\in P} x_P & \leq 1 \text{ for each arc } (u, v) \in A
\end{aligned}$$

The dual is as follows:

$$\begin{aligned}
\min \sum_{(u,v)\in E} & z_{uv} \\
z_{uv} & \geq 0 \text{ for each arc } (u, v) \in A \\
\sum_{(u,v)\in P} z_{uv} & \geq 1 \text{ for each path } P \in \mathcal{P}
\end{aligned}$$

For the maximum flow problem defined above, we define an *s-t cut* as a set S of nodes that contains s and does not contain t. An arc (u, v) is *in the cut* if it leaves S, i.e., u is in S and v is not. For a cut S, let $n(S)$ denote the number of edges leaving the cut. Note that every s-t cut gives an integer solution to the dual of the maximum flow problem by setting $z_{uv} = 1$ if e leaves set S, and 0 otherwise. All paths from s to t must leave set S at some point, hence they must contain at least one arc (u, v) with $z_{uv} = 1$. (Note that a path can leave S more than once, assuming it entered S again in between the two). Hence, this dual variable assignment is dual-feasible and has value $n(S)$.

Let z^* be an optimal dual solution. Recall that we defined $cost(s, w)$ for a node v mean the minimum, over all s to w paths P, of the sum of the optimal dual variable values for the edges on that path:

$$cost(s, w) = min_{s \to w \text{ path P}} \sum_{(u,v)\in P} z^*_{uv}.$$

We defined sets $S_\rho = \{v : cost(s, v) \leq \rho\}$, and showed that S_ρ defines an s-t cut for each $0 \leq \rho < 1$. We also proved the following inequality.

Lemma 1 *For each arc $(u, v) \in A$, we have that $cost(s, v) \leq cost(s, u) + z^*_{uv}$,*

We want to show that there exists a ρ^* such that $n(S_{\rho^*}) \leq \sum_{(u,v)\in A} z^*_{uv}$. By strong duality, the dual objective function is equal to the maximum flow value, and by weak duality, the maximum flow value is at most $n(S)$ for any $S \subseteq V$ as argued above. Thus $n(S_{\rho^*})$ equals the maximum flow value.

To find such a ρ^*, we pick one of the cuts S_ρ at random, by selecting ρ uniformly at random from the interval $[0, 1)$. The value of this cut is a random variable, and we will show that its expected value is at most $\sum_{(u,v)\in A} z^*_{uv}$, and hence there must exist at least one such cut that achieves this bound.

Lemma 2

$$E_\rho[n(S_\rho)] \leq \sum_{(u,v)\in A} z^*_{uv}.$$

Proof: We want to compute the expected number of edges leaving the cut S_ρ. To compute this expectation, first consider the probability that a given directed edge (u, v) leaves the randomly selected cut S_ρ. Edge (u, v) leaves S_ρ if and only if u is in S_ρ and v is not in S_ρ. This happens if and only if $cost(s, u) \leq \rho < cost(s, v)$. If $cost(s, u) < cost(s, v)$, then the probability that edge (u, v) leaves the randomly selected S_ρ is exactly $cost(s, v) - cost(s, u)$. Note that, by the lemma above, we get that $cost(s, v) - cost(s, u) \leq z^*_{uv}$; hence the probability that edge (u, v) leaves the selected set is at most z^*_{uv}.

Now, we compute the expected number of edges leaving the set. We can do this by introducing an indicator variable $I(u, v, \rho)$, which is equal to 1 if (u, v) leaves S_ρ, and is 0 otherwise. Then, we have that

$$n(S_\rho) = \sum_{(u,v)\in A} I(u, v, \rho).$$

By the linearity of expectation (that is, the expectation of a sum is the sum of the expectations), the expected value of $n(S_\rho)$ is equal to the sum, over all edges $(u, v) \in A$, of the expectation of $I(u, v, \rho)$. Since $I(u, v, \rho)$ is a 0-1 random variable, its expectation is equal to the probability that this variables is equal to 1; that is, the probability that edge (u, v) leaves the cut S_ρ, which is exactly what we bounded above. More precisely,

$$\begin{aligned} E_\rho[n(S_\rho)] &= E_\rho[\sum_{(u,v)\in A} I(u, v, \rho)] \\ &= \sum_{(u,v)\in A} E_\rho[I(u, v, \rho)] \\ &= \sum_{(u,v)\in A} \Pr[(u, v) \text{ in the cut } S_\rho] \\ &\leq \sum_{(u,v)\in A} z^*_{uv}, \end{aligned}$$

as desired. □

Since we know that the expected value of the cuts given our choice of ρ is at most the dual objective value, there must exist some ρ^* such that $n(S_{\rho^*})$ is at most the dual objective value and we are done.

From this point, we will take several lectures to build up to a proof of strong duality. We are going to start by building up some geometric and algebraic understanding of the feasible region.

We will consider the feasible region of a set of inequalities. Given a set of inequalities we define the feasible region as $P = \{x \in \Re^n : Ax \leq b\}$. We say that P is a *polyhedron.*

Our intuition from last time is that optimal solutions to linear programming problems occur at "corners" of the feasible region. What we'd like to do now is to consider formal definitions of the "corners" of the feasible region.

One idea is that a point in the polyhedron is a corner if there is some objective function that is minimized there and at no other point of P.

Definition 1 *$x \in P$ is a* vertex *of P if $\exists c \in \Re^n$ with $c^T x < c^T y, \forall y \neq x, y \in P$.*

Another idea is that a point $x \in P$ is a corner if there are no small perturbations of x that are in P.

Definition 2 *Let P be a convex set in $\Re^n$. Then $x \in P$ is an* extreme point *of P if x cannot be written as $\lambda y + (1-\lambda)z$ for $y, z \in P$, $y, z \neq x$, $0 \leq \lambda \leq 1$.*

It is interesting to note that because these definitions are generalized for all convex sets - not just polyhedra - a point could possibly be extreme but not be a vertex. One set of examples are the points on an oval where the line segments of the sides meet the curves of the ends.

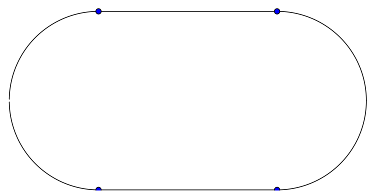

Figure 1: *Four extreme points in a two-dimensional convex set that are not vertices.*

A final possible definition is an algebraic one. We note that a corner of a polyhedron is characterized by a point at which several constraints are simultaneously satisfied. For any given x, let $A_=$ be the constraints satisfied with equality by x; (that is, a_j such that $a_j x = b_j$). Let $A_<$ be the constraints a_j such that $a_j x < b_j$.

Definition 3 *Call $x \in \Re^n$ a* basic solution *of P if $A_=$ has rank n. x is a* basic feasible solution *of P if it also lies inside P (so each constraint is either in $A_=$ or $A_<$).*

Since there are only a finite number of constraints defining P, there are only a finite number of ways to choose $A_=$, and if $rank(A_=) = n$ then x is uniquely determined by $A_=$. So there are at most $\binom{m}{n}$ basic solutions.

Theorem 3 *(**Characterization of Vertices**). Let P be defined as above. The following are equivalent:*

(1) x is a vertex of P.

(2) x is an extreme point of P.

(3) x is a basic feasible solution of P.

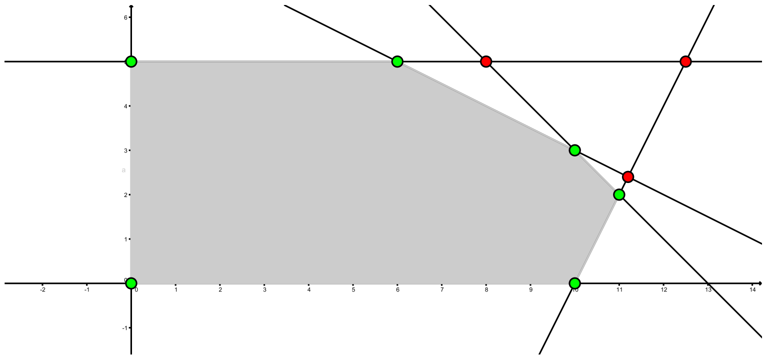

Figure 2: *A geometric representation of the nine basic solutions (green or red) and six basic feasible solutions (green) of an LP problem with six inequality constraints on two variables.*

Proof: We first prove that (1) $\Rightarrow$ (2). Let x be a vertex of P and suppose by way of contradiction that x is not an extreme point of P. Since x is a vertex, $\exists c \in \Re^m$ such that $c^T x < c^T y$ for all $y \in P, y \neq x$. Because x is not an extreme point, there exist $y, z \in P$, $y, z \neq x$, $0 \leq \lambda \leq 1$ such that $x = \lambda y + (1-\lambda)z$. Therefore $c^T x < c^T y$ and $c^T x < c^T z$. Thus

$$c^T x < \lambda c^T y + (1-\lambda) c^T z = c^T(\lambda y + (1-\lambda)z) = c^T x.$$

This gives us a contradiction, so x must be an extreme point.

We now prove (2) $\Rightarrow$ (3), by proving the contrapositive, that $\neg(3) \Rightarrow \neg(2)$. If x is not a basic feasible solution and $x \in P$, then the column rank of $A_=$ is less than n. Hence there is a direction vector $0 \neq y \in \Re^n$ such that $A_= y = 0$ (i.e. the columns of $A_=$ are linearly dependent). We want to show that for some $\varepsilon > 0$, $x + \varepsilon y \in P$ and $x - \varepsilon y \in P$. Then we will have shown that x can be written as a convex combination of two other points of P, since then $x = \frac{1}{2}(x+\varepsilon y) + \frac{1}{2}(x - \varepsilon y)$, which contradicts x being an extreme point. To show we can find the appropriate $\varepsilon > 0$, we first note that since $A_< x < b_<$, $b_< - A_< x > 0$, so we can choose a small $\varepsilon > 0$ such that $\varepsilon A_< y \leq b_< - A_< x$ and $-\varepsilon A_< y \leq b_< - A_< x$. Now to show that $x + \varepsilon y \in P$, we must show that $A_=(x+\varepsilon y) \leq b_=$ and $A_<(x + \varepsilon y) \leq b_<$. For the first inequality we have that

$$A_=(x + \varepsilon y) = A_= x + \epsilon A_= y = A_= x = b_=$$

since $A_= y = 0$. For the second inequality we have that

$$A_<(x+\varepsilon y) = A_< x + \epsilon A_< y \leq A_< x + (b_< - A_< x) = b_<$$

by our choice of ε. Showing that $x - \varepsilon y \in P$ is similar.

Finally, we prove (3) $\Rightarrow$ (1). Let $J = \{j : a_j^T x = b_j\}$. Set $c = -\sum_{j \in J} a_j^T$. Then

$$c^T x = \sum_{j \in J} a_j x = -\sum_{j \in J} b_j,$$

and for any $y \in P$,

$$c^T y = -\sum_{j \in J} a_j y \geq -\sum_{j \in J} b_j = c^T x$$

by the feasibility of y. Then it must be the case that $c^T y = c^T x$ only if $a_j y = b_j$ for all $j \in J$. Thus $A_= y = b_=$. However, since x is a basic feasible solution, $A_=$ has rank n, so that $A_= x = b_=$ has a unique solution x. Then if $c^T x = c^T y$, it must be the case that $x = y$. Hence we have that $c^T x = c^T y$ implies that $x = y$ and $c^T x \leq c^T y$ for all $y \in P$, and thus x is a vertex. $\square$