CS 683: Advanced Algorithms	**Lecture 2: Linear Programming Duality**
January 25th, 2001	**Scribes: Alex Druyan, Vladimir Dizhoor**

1 Fourier-Motzkin Algorithm

Recall from the previous lecture the following problem:

We are given a set of resources and customers. Each customer rents resource usage and each resource has some capacity T. The question is: how much revenue can we generate?

Let say that x_i is the rent time allocated to customer i, y_j is the amount of resource j that has been consumed and b_i is the capacity of resource i. We can define vectors $x = (x_1, x_2, \ldots, x_n)$ and $b = (b_1, b_2, \ldots, b_m) = (T, T, \ldots, T)$ from the given problem instance. Let $A_{ij} = 1$ if customer j requires resource i, and 0 otherwise. With these definitions the problem can be formulated as follows:

Maximize $c \cdot x$ subject to $Ax \leq b$, with $x \geq 0$, where c is a vector of size n consisting of just 1's.

Due to the Linear Programming Duality, the above question (the decision version) is equivalent to the following one:

Minimize $y \cdot b$ subject to $y \cdot A \geq c$, with $y \geq 0$ (c is as above).

First ask the decision version. When does the system $Ax \leq b$ have no solution? To answer it we need to determine whether or not a positive linear combination of the inequalities contained in the system above will yield $0 \leq -1$. Clearly, if such result is obtained then no solution to the system exists. We can, however, make a stronger statement:

Theorem 1.1 *(Farkas' Lemma): $Ax \leq b$ has no solution if and only if some non-negative linear combination of the inequalities of the system yields $0 \leq -1$.*

Consider a simple example:

$$-x_2 + x_1 \leq 0$$

$$-x_2 - x_1 \leq 0$$

$$x_2 \leq -1$$

If just the first two inequalities are in the system, then A solution clearly exists, however if all three are used then there is no solution to x. This is clear from the fact that adding the first two inequalities to twice the third inequality, yields $0 \leq -2$, or equivalently $0 \leq -1$.

The question "$\exists x? \ : \ Ax \leq b$" is in NP: we just exhibit such a vector x, and rely on a theorem (not proven here) that some such x can be represented using a number of bits that is polynomial in the size of the input. The theorem above, together with a similar bound on the bit representation, also shows that this question is in $co-NP$.

By representing the operations of linear combinations as a result of vector multiplication, we can restate the theorem as follows:

Theorem 1.2 *(Farkas' Lemma in matrix vector form): $Ax \leq b$ has no solution if and only if $\exists y \geq 0$ so that $yA = 0$ and $y \cdot b < 0$.*

Proof. (Constructive, but exponential)
"$\Longleftarrow$": Assume that $\exists y \geq 0$ so that $yA = 0$ and $y \cdot b < 0$. Assume $Ax \leq b$ has solution, then

$$0 > y \cdot b \geq y \cdot (Ax) = (yA) \cdot x = 0 \cdot x = 0$$

which is a contradiction, implying that $Ax \leq b$ has no solution.
"$\Longrightarrow$": Determining whether or not a system of inequalities has a solution is reminiscent of determining the same thing for a system of equations. Gauss' recursive method of successively eliminating variables using linear combination of rows accomplishes the latter task. Can we do something similar in case of inequalities? It turns out that a similar approach exists.

Here is an algorithm for "$\exists \mathbf{x}? \, : \, A\mathbf{x} \leq \mathbf{b}$" based on eliminating the first unknown variable x_1 and then proceeding recursively on $[x_2, x_3, \cdots, x_n]^T$. (Fourier-Motzkin elimination):

Initially, we are given a system of inequalities:

$$\mathbf{a}_1 \cdot \mathbf{x} \leq b_1$$
$$\mathbf{a}_2 \cdot \mathbf{x} \leq b_2$$
$$\vdots$$
$$\mathbf{a}_m \cdot \mathbf{x} \leq b_m$$

(Here $\mathbf{a}_j$ is the j'th row of A).

We can obviously rearrange the inequalities so that those inequalities which have a positive coefficient multiplying x_1 come first, those inequalities which have a negative coefficient multiplying x_1 follow and those inequalities which have a 0 multiplying x_1 come last. We then normalize the system of inequalities (divide those inequalities which have a nonzero coefficient multiplying x_1 by the absolute value of that coefficient). We thus arrive at a new system of inequalities (which is clearly equivalent to the original since no inequality was multiplied by a negative constant):

$$x_1 + \mathbf{a}'_1 \cdot \mathbf{x}' \leq b'_1$$
$$x_1 + \mathbf{a}'_2 \cdot \mathbf{x}' \leq b'_2$$
$$\vdots$$
$$x_1 + \mathbf{a}'_{k-1} \cdot \mathbf{x}' \leq b'_{k-1}$$
$$-x_1 + \mathbf{a}'_k \cdot \mathbf{x}' \leq b'_k$$
$$-x_1 + \mathbf{a}'_{k+1} \cdot \mathbf{x}' \leq b'_{k+1}$$
$$\vdots$$
$$-x_1 + \mathbf{a}'_l \cdot \mathbf{x}' \leq b'_l$$
$$0 + \mathbf{a}'_{l+1} \cdot \mathbf{x}' \leq b'_{l+1}$$
$$\vdots$$
$$0 + \mathbf{a}'_m \cdot \mathbf{x}' \leq b'_m$$

Denote the last $m - l$ inequalities (i.e., the inequalities with the 0 coefficient multiplying x_1) by (**). The coefficients of x_1 in the new system are $+1$, -1 or 0. If all nonzero coefficients share the same sign (for instance +) then we can proceed recursively to solve for x' and then make x_1 as negative as necessary. If both signs are encountered then we will find inequalities:

$$+x_1 + \mathbf{a}'_i \cdot \mathbf{x}' \leq b'_i \;\rightarrow\; x_1 \leq b'_i - \mathbf{a}'_i \cdot \mathbf{x}'$$
$$-x_1 + \mathbf{a}'_j \cdot \mathbf{x}' \leq b'_j \;\rightarrow\; \mathbf{a}'_j \cdot \mathbf{x}' - b'_j \leq x_1$$

This yields the following condition on bounds of the region where x_1 is to be found:

$$\mathbf{a}'_j \cdot \mathbf{x}' - b'_j \leq b'_i - \mathbf{a}'_i \cdot \mathbf{x}'$$

So, a valid solution requires

$$(\mathbf{a}'_i + \mathbf{a}'_j) \cdot \mathbf{x}' \leq b'_i + b'_j \tag{1}$$

Thus we can form a new system of inequalities consisting of inequality (1) for each i, j pair such that

$$i \leq i \leq k, \quad k+1 \leq i \leq l$$

and inequalities (**).

Claim: the new system has a solution if and only if the old one does.

Proof: one direction is obvious: since the new system is a nonnegative combination of inequalities of the old system, the new inequalities are all implied by the original inequalities. The other direction is proven by the following argument. Having found x', look at the smallest $b'_i - \mathbf{a}'_i \cdot x$ and largest $\mathbf{a}'_j \cdot x - b'_j$. Since the solution to inequalities (1) exists, it must be the case that there is some space between the two bounds. By placing x_1 somewhere in between those bounds we clearly get a solution to the original system. ■

This algorithm is not very efficient, however. An actual algorithm can be very "nasty" since it is "doubly-exponential" but can be polynomial in practice.

2 When the system has no solution

When there is no solution, here is a way to derive $0 \leq -1$. This can be done recursively (proof by induction):

Base case: when $-b < a$ so that $a + b < 0$, and the following two inequalities hold

$$+x_n \leq a$$

$$-x_n \leq b$$

then there is no solution, since then

$$0 \leq a + b < 0$$

Recall that we made up this new system by just adding together pairs of rows of the original system. If the "new system" on $n-1$ variables has no solution, then we can take non-negative combinations of the "new rows" to get $0 \leq -1$. But each new row is either in the original system, or is a sum of two rows in the original system and that gives us a non-negative combination of original rows.

3 A version of duality

We assume that the set $\{x' \mid Ax' \leq b\}$ is non-empty, i.e., $\{x' \mid Ax' \leq b\} \neq \emptyset$, so that we will not have to bother with dealing with the boundary cases, and can safely assume an existence of at least one element in that set.

The duality theorem states that

$$\max\{c \cdot x \mid Ax \leq b\} = \min\{y \cdot b \mid yA = c, \ y \geq 0\}$$

How do we prove now that some γ cannot be achieved by the above maximum? We just need to show that the system of inequalities

$$Ax \leq b$$

$$c \cdot x \geq \gamma$$

has no solution. But this is equivalent to the system

$$Ax \leq b$$

$$-c \cdot x \leq -\gamma$$

where the last inequality of the previous inequality system has been simply multiplied by -1, which does not change the system. Thus, there is no solution if and only if $\exists y \geq 0$, which we write as $y = [y', y_0]$, such that

$$[y', y_0][A, -c]^T = 0$$

$$[y', y_0][b, -c]^T < 0$$

or equivalently

$$y'A - y_0 C = 0$$

$$y'b - y_0 \gamma < 0$$

Note that y_0 cannot be zero, since if it were zero, then $y'A = 0$, $y'b < 0$ implying that the original inequality had no solution, so that the set $\{x' \mid Ax' \leq b\}$ is empty, but this contradicts the initial assumption that there was at least one element in that set.

So $y_0 > 0$. Re-scaling, we assume that $y_0 = 1$. From this, it follows that $y'A = c$, $y'b < \gamma$ $(y' \geq 0)$. Can show that if

$$\gamma^* = \min\{y'b \mid y'A = c\},\ y' \geq 0$$

then

$$\gamma^* = \max\{cx \mid Ax \leq b\}$$

4 A second version of duality

$$\gamma^* = \max\{cx \mid Ax \leq b,\ x \geq 0\}$$

$$\Longleftrightarrow$$

$$\gamma^* = \max\{cx \mid Ax \leq b,\ -Ix \leq 0\}$$

By the previous duality result, the above is also equal to

$$\gamma^* = \min\{yb \mid yA \geq c,\ y \geq 0\}$$

5 Set Packing

Now, lets get back to our very first problem of "Set Packing". Lets consider the discrete version of the problem. Given a collection of sets $S_1, S_2, \ldots, S_n$, one is asked to find a maximum size collection of mutually disjoint sets. This is similar to our original resource allocation problem, although discrete, with extra integer conditions placed on it.

Consider the following: Suppose that in a given instance of the Set-Packing problem there are 3 elements so that every set S_i contains at least one of them. Then clearly, no collection of more than 3 sets can possibly be mutually disjoint, and thus, the maximal size of the collection of mutually disjoint sets cannot exceed 3. This is an obstacle to the Set-Packing problem.

A "hitting set" (of size k) is a collection of k elements so that each S_i contains at least one of them. Thus, we see that small hitting sets are an obstacle to Set-Packing.

Clearly, if there is a hitting set of size k then the maximal collection of disjoint sets has size $\leq k$. Is this claim an if and only if? Consider the set packing problem with sets $\{1, 2\}$, $\{1, 3\}$, and $\{2, 3\}$. There is no hitting set of size 1, but there are no two mutually disjoint sets.

The if and only if that we had initially was simply a fractional version of Set-Packing.

5.1 Fractional Set-Packing

Put a weight ≥ 0 on each set so that the total weight of the sets containing any element is ≤ 1. Here we maximize the total weight.

5.2 Fractional Hitting-Set

Put a value ≥ 0 on each element so that the total value of elements in each set is ≥ 1. Here we minimize the total value.