COMPSCI 330: Design and Analysis of Algorithms 11/11/2014

Lecture #21

Lecturer: Debmalya Panigrahi *Scribe: Samuel Haney*

1 Overview

In this lecture, we discuss linear programming. We first show that the various forms of linear programs are all equivalent. We give an example showing how to express a problem as a linear program. We discuss separation oracles and duality.

2 Forms of Linear Programs

In this section, we define common linear program forms, and show that these are equivalent.

Definition 1. *A linear program is in* ***standard form*** *if it has the following structure.*

$$\begin{aligned} \textit{maximize} \quad & \mathbf{c}^{\mathbf{T}}\mathbf{x} \\ \textit{subject to} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \overrightarrow{0} \end{aligned}$$

We use bold lowercase letters to denote vectors, and bold uppercase letters to denote matrices.

Definition 2. *A linear program is in* ***general form*** *if it has the following structure.*

$$\begin{aligned} \textit{minimize/maximize} \quad & \mathbf{c}^{\mathbf{T}}\mathbf{x} \\ \textit{subject to} \quad & \mathbf{A}_1\mathbf{x} \leq \mathbf{b}_1 \\ & \mathbf{A}_2\mathbf{x} = \mathbf{b}_2 \\ & \mathbf{A}_3\mathbf{x} \geq \mathbf{b}_3 \\ & \mathbf{x}_1 \geq \overrightarrow{0} \\ & \mathbf{x}_2 = \overrightarrow{0} \\ & \mathbf{x}_3 \leq \overrightarrow{0} \end{aligned}$$

Definition 3. *A linear program is in* ***slack form*** *if it has the following structure.*

$$\begin{aligned} \textit{maximize} \quad & \mathbf{c}^{\mathbf{T}}\mathbf{x} \\ \textit{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \overrightarrow{0} \end{aligned}$$

Claim 1. *Standard form, general form, and slack form LPs are all equivalent.*

We prove Claim 1 with a series of lemmas which show that every form can be converted into every other form. First, observe that programs in standard form or slack form are already in general form, so this direction is done.

Lemma 1. *A linear program in general form can be transformed into standard form.*

Proof. We transform minimization LPs to maximization by negating the objective:

$$\text{minimize } \mathbf{c}^{\mathbf{T}}\mathbf{x} \quad \longrightarrow \quad \text{maximize } -\mathbf{c}^{\mathbf{T}}\mathbf{x}. \tag{1}$$

We also transform "$\geq$" constraints to "$\leq$" constraints using negation:

$$\mathbf{A}_3\mathbf{x} \geq \mathbf{b}_3 \quad \longrightarrow \quad -\mathbf{A}_3\mathbf{x} \leq -\mathbf{b}_3. \tag{2}$$

We transform equality constraints to inequality constraints by writing two inequalities:

$$\mathbf{A}_2\mathbf{x} = \mathbf{b}_2 \quad \longrightarrow \quad \begin{array}{l} \mathbf{A}_2\mathbf{x} \geq \mathbf{b}_2, \text{ and} \\ \mathbf{A}_2\mathbf{x} \leq \mathbf{b}_2. \end{array} \tag{3}$$

From these inequalities, we use transformation 2 to transform the remaining constraints to "$\leq$" constraints. We transform the variable equality constraints to inequality constraints:

$$\mathbf{x}_2 = 0 \quad \longrightarrow \quad \begin{array}{l} \mathbf{x}_2 \geq 0, \text{ and} \\ \mathbf{x}_2 \leq 0. \end{array} \tag{4}$$

Finally, $\mathbf{x}_3 \leq 0$ constraints are of the form $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, and are therefore allowed (the corresponding entries in $\mathbf{b}$ will be 0, and the entries in $\mathbf{A}$ will be 1). □

Lemma 2. *A linear program in standard form can be transformed into slack form.*

Before we give the proof, note that Lemma 1 and Lemma 2 complete the proof of Claim 1. We can convert from general to slack form by composing Lemma 1 with Lemma 2. Now, we give a proof of Lemma 2

Proof. Constraints in standard form have structure

$$\mathbf{A}\mathbf{x} \leq \mathbf{b}.$$

We introduce variables $\mathbf{s}$, and transform the above constraints as follows:

$$\mathbf{A}\mathbf{x} \leq \mathbf{b} \quad \longrightarrow \quad \begin{array}{l} \mathbf{A}\mathbf{x} + \mathbf{s} = \mathbf{b}, \text{ and} \\ \mathbf{s} \geq 0. \end{array} \tag{5}$$

A solution $\mathbf{x}$ to initial LP is feasible if and only if there exist a setting of variables $\mathbf{s}$, along with the setting of $\mathbf{x}$, such that the transformed LP is feasible. □

3 Minimum s-t Cut as an LP

We first review the definition of the minimum s-t cut problem. Let $G = (V,E)$ be a graph, and let $s,t \in V$. Let edge e have capacity u_e. Our goal is to find a cut $(S,\bar{S})$ such that

$$\sum_{e \in (S,\bar{S})} u_e$$

is minimized, and $s \in S$ and $t \in \bar{S}$. We formulate this problem as an integer program:

$$\begin{aligned} \text{minimize} \quad & \sum_{e \in E} u_e x_e \\ \text{subject to} \quad & \sum_{e \in P} x_e \geq 1 \quad \forall \text{paths } P \text{ from } s \text{ to } t \\ & x_e \in \{0,1\} \quad \forall e \in E \end{aligned}$$

This is an integer program since values of x_e must be integral. We relax our formulation to an LP.

$$\begin{aligned} \text{minimize} \quad & \sum_{e \in E} u_e x_e \\ \text{subject to} \quad & \sum_{e \in P} x_e \geq 1 \quad \forall \text{paths } P \text{ from } s \text{ to } t \\ & x_e \geq 0 \qquad \forall e \in E \end{aligned}$$

We call this the fractional minimum s-t cut problem. Can we solve this LP? There is a problem: the number of constraints of the LP is the number of paths from s to t, which could be exponential. In the following section, we show how it is sometimes possible to solve LPs with exponential constraints.

4 Separation Oracles

Definition 4. *A **separation oracle** is a polynomial algorithm, which given an (possibly exponential sized) LP and a solution vector (i.e. a setting of the variables)* $\mathbf{x}$ *outputs*

1. *YES if* $\mathbf{x}$ *is feasible for the LP (i.e.* $\mathbf{x}$ *satisfies all the constraints), or*
2. *a violated constraint otherwise.*

Theorem 3. *Any LP with a separation oracle can be solved in polynomial time.*

We won't give a proof of Theorem 3

Example . Consider the following LP.

$$\begin{aligned} \text{maximize} \quad & x \\ \text{subject to} \quad & x + y \leq 10 \\ & y \geq 5 \end{aligned}$$

Given solution vector $(5,6)$, a separation oracle would output constraint $x+y \leq 10$, since $(5,6)$ violates this constraint.

Note that separation oracles are trivial for polynomial sized LPs. In this case, the oracle can check every constraint in the LP to see if each is satisfied. If the LP has exponentially many constraints, the oracle cannot check every constraint.

Example . Is there a separation oracle for fractional minimum s-t cut? That is, given $G = (V,E)$ and given x_e for all e, is there a path P from s to t such that

$$\sum_{e \in P} x_e < 1?$$

Our constraints are violated if and only if such a path exists. However, finding whether such a path exists is just the shortest path problem! As we have seen, the shortest path problem can be solved in polynomial time. Therefore, this is a separation oracle for fractional minimum s-t cut, and so by Theorem 3 the LP is solvable in polynomial time.

5 LP Duality

In this section, we give a motivating example for LP duality. We will discuss duality more formally in the next lecture. Consider the following LP.

$$\begin{array}{rl} \text{minimize} & 10x + 10y \\ \text{subject to} & x + 3y \geq 4 \\ & 2x + y \geq 5 \\ & x \geq 0 \\ & y \geq 0 \end{array}$$

It is not immediately clear what the solution to this LP is. Instead of trying to find the optimal solution, we try to find a lower bound on the optimal. We multiply each of the first two constraints by some number, and add them.

$$\begin{array}{rl} & a \cdot (x + 3y \geq 4) \\ + & b \cdot (2x + y \geq 5) \\ \hline \end{array} \tag{6}$$

If we set $a = 2$ and $b = 3$, we get

$$\begin{array}{rl} & 2x + 6y \geq 8 \\ + & 6x + 3y \geq 15 \\ \hline & 8x + 9y \geq 23. \end{array}$$

This is significant because the objective is larger than the left side of this constraint. That is,

$$10x + 10y \geq 8x + 9y \geq 23.$$

Therefore, the optimal value of the objective, $10x + 10y$, is at least 23. However, which numbers a and b should we choose to get the best lower bound? Keeping a and b in Equation 6 gives

$$\begin{array}{rlll} & ax & +3ay & \geq 4a \\ + & 2bx & +by & \geq 5b \\ \hline & (a+2b)x & +(3a+b)y & \geq 4a + 5b. \end{array}$$

We want to maximize the right side of this constraint, $4a+5b$ to get the highest lower bound possible. Additionally, the left side of the constraint should not be more than the objective. We can write this as follows.

$$\begin{aligned} \text{maximize} \quad & 4a+5b \\ \text{subject to} \quad & a+2b \leq 10 \\ & 3a+b \leq 10 \\ & a \geq 0 \\ & b \geq 0 \end{aligned}$$

This is just another LP! We call this LP the dual, and the original LP the primal. The dual LP has one constraint for each variable in the primal, and one variable for each constraint in the primal. A principle called *weak duality* which we will discuss more formally in the next lecture, says that the maximum dual LP objective is at most the minimum primal LP objective. In fact, *strong duality* says that these two quantities are always equal.