**18.409 The Behavior of Algorithms in Practice** March 19, 2002

Lecture 11

*Lecturer: Dan Spielman* *Scribe: José Correa*

# Linear Programming

We now start with the study of smoothed analysis for linear programming. In the forthcoming lectures we shall cover three methods for LP:

1.- Elementary/Naive/Stupid methods (this lecture).

2.- Simplex method (part in this lecture).

3.- Interior point method.

## Introduction

Given $a_1, \ldots, a_n \in \mathbb{R}^d$ and $c \in \mathbb{R}^d$, Dan's favorite linear program is the following

$$
\begin{array}{lll}
\text{[LP1]} \ \max & \alpha \\
\text{s.t.} & \alpha c \in \text{ch}(a_1, \ldots, a_n), \\
& \alpha \geq 0.
\end{array}
$$

where $\text{ch}(a_1, \ldots, a_n)$ denotes the convex hull defined by the vectors $a_1, \ldots, a_n$. We know that $c \in \text{ch}(a_1, \ldots, a_n)$ if and only if there exists $y_1, \ldots, y_n \geq 0$ such that $\sum_{i=1}^n y_i = 1$ and $c = \sum_{i=1}^n y_i a_i$. Hence we can write [LP1] in an equivalent form

$$
\begin{array}{lll}
\text{[LP2]} \ \max & \alpha \\
\text{s.t.} & \exists y_1 \ldots y_n \geq 0 \\
& \sum_{i=1}^n y_i = 1, \ \sum_{i=1}^n y_i a_i = \alpha c.
\end{array}
$$

By letting $y_i' = \frac{y_i}{\alpha}$, we have that $y_i' \geq 0$, $\sum_{i=1}^n y_i' = \frac{1}{\alpha}$ and $c = \sum_{i=1}^n y_i' a_i$. Then, maximizing $\alpha$ is the same as minimizing $\sum_{i=1}^n y_i'$, and hence [LP2] becomes

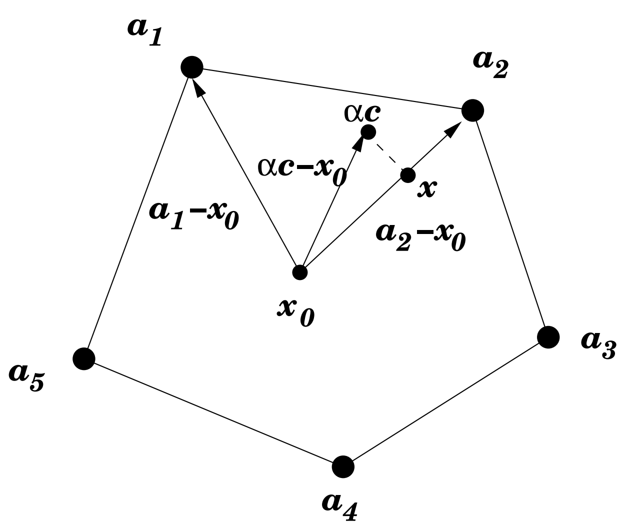
$$\begin{aligned} \text{[LP3]} \ \min \quad & \sum_{i=1}^n y_i' \\ \text{s.t.} \quad & \sum_{i=1}^n y_i' a_i = c, \\ & y_i' \geq 0. \end{aligned}$$

## 1.- Von Neumann's algorithm for LP

We know see an elementary algorithm to solve [LP1]. The algorithm will use binary search on $\alpha$. And it will call a subroutine which decides whether a given $\alpha c$ belongs to $\mathrm{ch}(a_1, \ldots, a_n)$, this is called the *decision problem*.

To solve the decision problem Von Neumann's algorithm proceeds as follows:

- Take $x_0 \in \mathrm{ch}(a_1, \ldots, a_n)$, say $x_0 = \frac{1}{n} \sum_{i=1}^n a_i$.
- Choose $i$ maximizing $\langle(\alpha c - x_0), (a_i - x_0)\rangle$ ($i = 2$ in the figure below).
- Find the point $x$ from $x_0$ to $a_i$ (i.e. $x \in \mathrm{ch}(x_0, a_i)$) closest to $\alpha c$.
- $x_0 = x$ and repeat.




This procedure converges since whenever $\alpha c \in \mathrm{ch}(a_1, \ldots, a_n)$ we have $\max_i\{\langle(\alpha c - x_0), (a_i - x_0)\rangle\} \geq 0$ for some $i$. Otherwise, $\langle(\alpha c - x_0), (a_i - x_0)\rangle < 0$ for all $a_i$. This implies that there is an hyperplane separating $x_0$ form $\{a_1, \ldots, a_n\}$, from where $\alpha c \notin \mathrm{ch}(a_1, \ldots, a_n)$.

**Theorem 1 (Dantzig).** *Von Neumann's Algorithm obtains a point $x_0$ such that $||x_0 - \alpha c|| < \epsilon$ in*

$$\frac{4 \max\{||\alpha c||, \max_i ||a_i||\}}{\epsilon^2}$$

*iterations.*

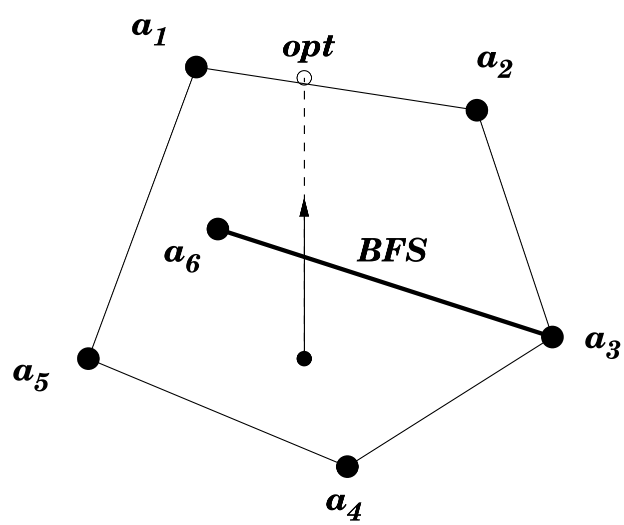
**Theorem 2 (Freund-Epelman).** *Von Neumann's Algorithm obtains a point $x_0$ such that $||x_0 - \alpha c|| < \epsilon$ in*

$$\frac{8 \max\{||\alpha c||, \max_i ||a_i||\}}{r^2} \log\left(\frac{||x_0 - \alpha c||}{\epsilon}\right)$$

*iterations. Where $r = \mathrm{distance}(||\alpha c||, \mathrm{boundary}(\mathrm{ch}(a_1, \ldots a_n)))$. $r$ is a condition number of the linear program.*

## 2.- The Simplex method

Consider again the linear program [LP1]. A basic feasible solution of [LP1] is collection of $d$ points, $B \subset \{a_1, \ldots, a_n\}$ ($|B| = d$) such that $\alpha c \in \mathrm{ch}(B)$ for some $\alpha \geq 0$.




The simplex method proceeds as follows:

- Find a Basic feasible solution $B$.
- Find point, say $a$, in $\{a_1, \ldots, a_n\}$ above (with respect to $c$) the hyperplane $\mathrm{ch}(B)$.
- Remove one point, $b$, from $B \cup \{a\}$ so that $B' = B \cup \{a\} \setminus \{b\}$ is a basic feasible solution.
- $B = B \cup \{a\} \setminus \{b\}$ and repeat.

**Initialization:** Plant a Basic feasible solution. i.e. put $d$ points very close to the origin (say at distance $\epsilon$), so that they are not involved in an optimal solution. This is also known as the big $M$ method since, $\min\{\sum_{i=1}^n y_i : \sum_{i=1}^n y_i a_i = c,\ y_i \geq 0\}$ can be written as

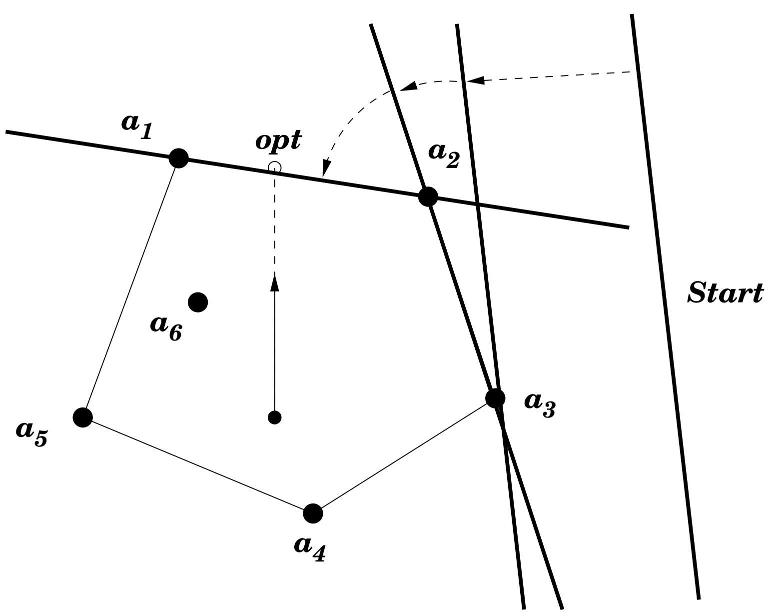
$$\begin{aligned} \text{[LP3]} \ \min \quad & \sum_{i=1}^{n} y_i + M \sum_{j=1}^{d} z_j \\ \text{s.t.} \quad & \sum_{i=1}^{n} y_i a_i + \sum z_j e_j = c, \\ & y_i \geq 0. \end{aligned}$$

Which again, by letting $M = 1/\epsilon$, is equivalent to

$$\begin{aligned} \text{[LP3]} \ \min \quad & \sum_{i=1}^{n} y_i + \sum_{j=1}^{d} z_j \\ \text{s.t.} \quad & \sum_{i=1}^{n} y_i a_i + \sum z_j (\epsilon e_j) = c, \\ & y_i \geq 0, \end{aligned}$$

and this is exactly the initialization method described above.

**Duality:** Another way of looking at Dan's linear program [LP1] is the following. Find a plane $H$ and the minimum $\alpha$ such that, $\alpha c \in H$ and $\{a_1, \ldots, a_n\}$ are beneath $H$.




Since $H_x = \{a : \langle a, x \rangle = 1\}$ is the plane normal to a given vector $x$. The above problem, know as the *dual*, can be stated as

$$\begin{aligned} \text{[Dual-LP1]} \quad \min \quad & \alpha \\ \text{s.t.} \quad & \langle \alpha c, x \rangle = 1 \\ & \langle a_i, x \rangle \leq 1 \text{ for all } i. \end{aligned}$$

Now $\langle c, x \rangle = 1/\alpha$, hence the problem is simply

$$\begin{aligned} \text{[Dual-LP3]} \quad \max \quad & \langle c, x \rangle \\ \text{s.t.} \quad & \langle a_i, x \rangle \leq 1 \text{ for all } i. \end{aligned}$$

We can conclude the following result.

**Theorem 3.** *If primal [LP1] has a solution, then the dual [Dual-LP1] has the same solution.*