CSCI-B609: A Theorist's Toolkit, Fall 2016	Nov 3

Lecture 19: Solving Linear Programs

Lecturer: Yuan Zhou *Scribe: Chao Tao*

1 Recap

In Leture 18, we have talked about **Linear Programming (LP)**. LP refers to the following problem. We are given an input of the following m constraints (inequalities):

$$K \subset \mathbb{R}^n = \begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \le b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \le b_m \end{cases} \tag{1}$$

where $a_{ij}, b_i \in \mathbb{Q}$, for $i \in [m], j \in [n]$. Our goals are 1) to determine whether a solution exists, i.e., $K \neq \emptyset$ and 2) to maximize $c^T x$ for some $c \in \mathbb{Q}^n$ such that $x \in K$, if $K \neq \emptyset$. We will focus on goal 1). For goal 2), we can add one more constraint $c^T x \ge \text{opt}$ to K and binary search for opt.

2 Solving Linear Programs

Actually, besides (1), LP also has other forms. Notice that for each x_i, we can rewrite it to $x_i = x_i^+ - x_i^-$ where $x_i^+, x_i^- \ge 0$. Therefore, after doing this transformation, (1) will become the following **Standard Form**:

$$K = \begin{cases} Ax \le b \\ x \ge 0. \end{cases} \tag{2}$$

Let $A^{(i)}$ denote the i-th row of A. Then we can find that for each row $A^{(i)}$, $A^{(i)} \cdot x \le b_i \Leftrightarrow A^{(i)} \cdot x + s_i = b_i, s_i \ge 0$. After adding s_i's, (2) will become the following **Equational Form**:

$$K = \begin{cases} A'(x,s)^T = b \\ x \ge 0,\ s \ge 0 \end{cases} \text{ i.e. } K = \begin{cases} Ax = b \\ x \ge 0 \end{cases} \tag{3}$$

Therefore, it suffices to solve (3) in order to solve (1).

Definition 1. *Suppose* $b = \frac{p}{q} \in \mathbb{Q}$ *where* $p, q \in \mathbb{Z}$*. Define* $|b\rangle = |p\rangle + |q\rangle = \lceil \log_2 |p| \rceil + \lceil \log_2 |q| \rceil$.

Theorem 1. *Given* $K \subset \mathbb{R}^n = \begin{cases} Ax = b \\ x \geq 0 \end{cases}$*, let* $L = |A\rangle + |b\rangle = \sum_{ij} |a_{ij}\rangle + \sum_i |b_i\rangle$ *be the size of input. If* $K \neq \emptyset$*, there exists* $x \in K$ *such that* $|x\rangle = poly(L)$.

Proof. If we consider the geometric meanings of our constraints, then the set of solutions for $Ax = b$ is an affine linear subspace in $\mathbb{R}^n$. Suppose its dimension is m. We will prove this theorem via mathematical induction on m.

If $m = 0$, K is a single point and $Ax = b$ gives a solution x in K, which can be found in polynomial time by Gaussian Elimination. And this proves that $|x\rangle = \text{poly}(L)$. If $m = 1$, $Ax = b$ defines a line. The line can't be parallel to all hyperplanes defined by $x_i = 0$. It means that there exists a $i \in [n]$ such that $\begin{cases} Ax = b \\ x_i = 0 \end{cases}$ gives a solution x with $|x\rangle = \text{poly}(L)$. If $m = 2$, $Ax = b$ defines a 2-dimensional hyperplane. Therefore, there exists $i \in [n]$ such that $x_i = 0$ intersects with the plane. And $\begin{cases} Ax = b \\ x_i = 0 \end{cases}$ defines a line. And we can use former approach when $m = 1$ to find a solution x with $|x\rangle = \text{poly}(L)$. Generally, for dimension $m > 2$, we can reduce it to the case when dimension is $m - 1$. □

Corollary 2. *Let* $K \subset \mathbb{R}^n = \begin{cases} Ax = b \\ x \geq 0 \end{cases}$*. If* $K \neq \emptyset$*, there exists a bounding box* $B = \{x : \forall i, 0 \leq x_i \leq B_i\}$ *such that* $K \cap B \neq \emptyset$ *with* $|B\rangle = \sum_i |B_i\rangle = poly(L)$.

Definition 2. *LP' is the problem that given* $K \subset \mathbb{R}^n = \begin{cases} Ax \leq b \\ x_i \geq 0 \end{cases}$*, our goal changes to the following: if* $vol(K) \geq V$*, then output* $x \in K$ *where* $V = 2^{-poly(L)}$ *otherwise output nothing.*

Theorem 3. *If we can solve LP' in polynomial time, we can also solve LP in polynomial time.*

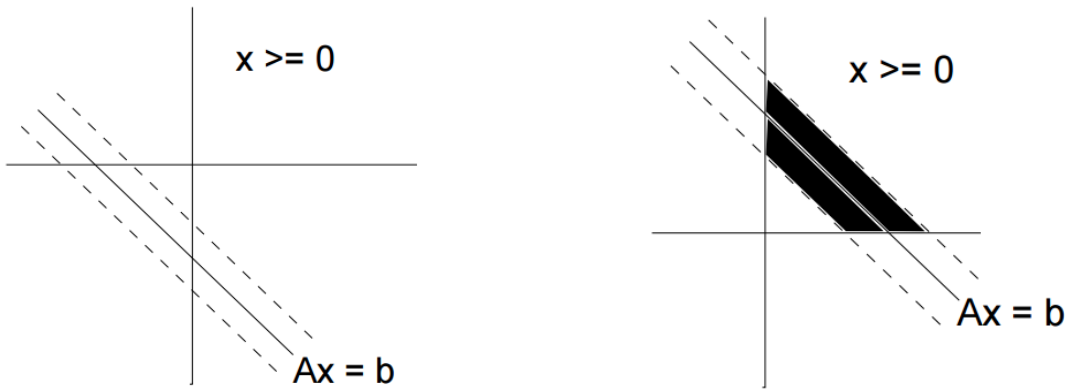
Remark 3.1. *LP' is a more relaxed (easier) task than LP.*

Proof Sketch. If $K = \emptyset$, let γ be the distance from $\{x : Ax = b\}$ to $\{x \geq 0\}$. One can show that $\gamma \geq 2^{-\text{poly}(L)} = \gamma_0$. Let $K' = \{x : -\gamma_0 \leq Ax - b \leq \gamma_0, x \geq 0\}$. We will get $K' = \emptyset$.

If $K \neq \emptyset$, K' has volume at least $\left(\frac{\gamma_0}{2^{\text{poly}(L)}}\right)^n = 2^{-\text{poly}(L)}$. Therefore, if LP' is solvable in polynomial time, we can relax K to K'. After that, we can trigger the LP' solver.

Here is a geometric view in $\mathbb{R}^2$.

□



(a) $K = \emptyset$ (b) $K \neq \emptyset$

Figure 1: A geometric view in $\mathbb{R}^2$

2.1 The Ellipsoid Method for LP

In this subsection, we will introduce a concrete method for solving LP. The high level idea is that we maintain an ellipsoid that contains K. We can assume $vol(K) \geq 2^{-\text{poly}(L)} = V$ by Theorem 3. Here is the algorithm.

Algorithm 1 The Ellipsoid Algorithm

Input: $K \subset \mathbb{R}^n = \begin{cases} Ax \leq b \\ x \geq 0 \end{cases}$.

Initialization: Let E_0 be the smallest ellipsoid containing the bounding box B defined in Corollary 2. Set $i = 1$.

while TRUE **do**

  Let x be the center of E_{i-1}.

  if $x \in K$ **then**

    Output: x.

  else

    Find an arbitrary linear constrain that is violated namely $a \cdot x \leq b$. Let E_i be the smallest ellipsoid containing $E_{i-1} \cap \{x : a \cdot x \leq b\}$.

  if $vol(E_i) < V$ **then**

    Output: NO SOLLUTION.

  $i = i + 1$.

The following graph gives a geometric view of picking E_1.

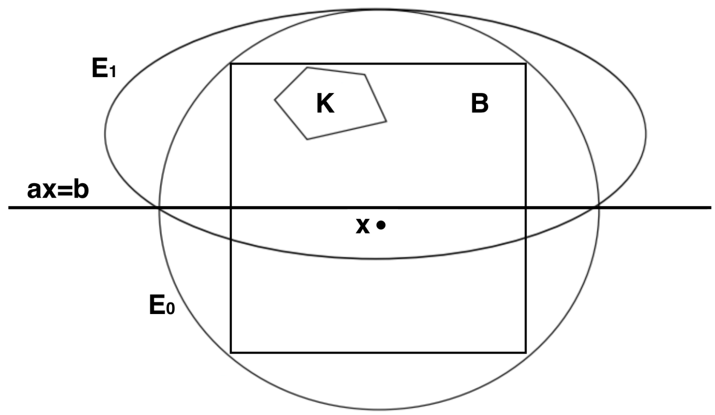


Figure 2: A geometric view of picking E_1

If a solution is returned, it must be a feasible solution of K. If NO SOLLUTION returned, $vol(K)$ must be $< V = 2^{-\text{poly}(L)}$ since $K \subset E_i$ for any i. Therefore, we only need to bound the number of iterations performed.

Claim 4. *For every* $i = 1, 2, 3, \ldots,\ \frac{vol(E_i)}{vol(E_{i-1})} \leq 1 - \frac{1}{3n}$.

Proof. Since every ellipsoid can be obtained via invertible linear transformations from a unit ball, and the volume is preserved upon a scaling factor (the determinant of the linear transformation matrix), we can assume w.l.o.g. that E_{i-1} is the unit ball.

Now the worst case for the separating hyperplane is to go through the origin. Assume w.l.o.g. that it is $x_1 \geq 0$.

Let E_i be centered at $(t, 0, \ldots, 0)$ $(t < 1/2)$. Let the semi-axis along x_1 be $(1 - t)$ and semi-axis along other directions be s, i.e. E_i satisfies the following equation:

$$\frac{(x_1 - t)^2}{(1-t)^2} + \frac{x_2^2}{s^2} + \cdots + \frac{x_n^2}{s^2} \leq 1.$$

To contain the half ball $E_{i-1} \cap \{x : x_1 \geq 0\}$ in E_i, we only need $(0, 1, 0, \ldots, 0) \in E_{i-1}$, i.e.

$$\frac{t^2}{(1-t)^2} + \frac{1}{s^2} \leq 1.$$

Here is the geometric view of above procedure.

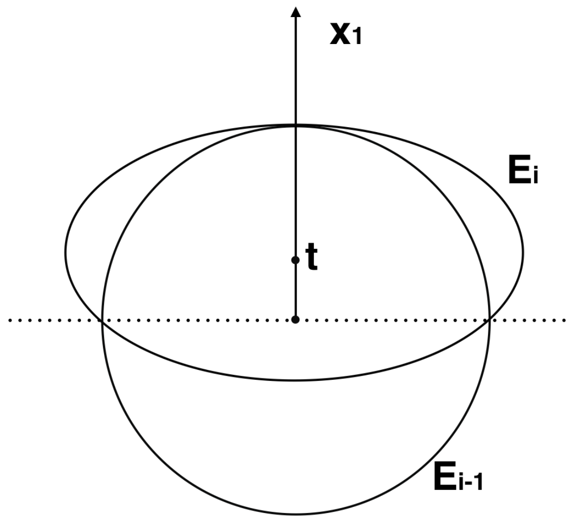


Figure 3: A geometric view of computing E_i

From this inequality, we can get $s^2 \geq \frac{(1-t)^2}{1-2t}$. Set $s^2 = \frac{(1-t)^2}{1-2t}$ and $t = \frac{1}{2n}$. We have

$$\begin{aligned}
\frac{vol(E_i)}{vol(E_{i-1})} &= (1-t) \cdot s^{n-1} \\
&= \left(\frac{(1-t)^2}{1-2t}\right)^{(n-1)/2} (1-t) \\
&= \left(\frac{1-1/n+1/(4n^2)}{1-1/n}\right)^{(n-1)/2} \left(1-\frac{1}{2n}\right) \\
&= \left(1+\frac{1}{4n(n-1)}\right)^{(n-1)/2} \left(1-\frac{1}{2n}\right) \\
&\leq \left(e^{\frac{1}{4n(n-1)}}\right)^{(n-1)/2} e^{-\frac{1}{2n}} \\
&= e^{-\frac{3}{8n}} \\
&\leq 1-\frac{1}{3n}.
\end{aligned}$$

In general, if E_i is not the unit ball, let σ be the invertible linear transformation such that $\sigma(\text{unit ball}) = E_i$. Let $\det(\sigma)$ be the determinant of the matrix associated with σ. Let $a' \cdot x \leq b'$ be the half-space obtained by $\sigma^{-1}(\{a \cdot x \leq b\})$. Let E_i' be the smallest ellipsoid containing unit ball $\cap \{a' \cdot x \leq b'\}$. By previous discussion:

$$\frac{vol(E_i')}{vol(\text{unit ball})} \leq 1-\frac{1}{3n}.$$

Let $E_i = \sigma(E'_i)$. We know that E_i contains $E_{i-1} \cap \{a \cdot x \le b\}$ since σ is bijective. Finally, we will have

$$\frac{vol(E'_i)}{vol(\text{unit ball})} = \frac{\det(\sigma) \cdot vol(E'_i)}{\det(\sigma) \cdot vol(\text{unit ball})} \le 1 - \frac{1}{3n}.$$

□

Claim 5. *The Ellipsoid Algorithm terminates within poly(L) iterations.*

Proof. Suppose the algorithm does not terminate after i iterations. Then we must have $vol(E_i) \ge V = 2^{-\text{poly}(L)}$. Also by selection of E_0, we have $vol(E_0) \le 2^{O(n)} \cdot vol(B) \le 2^{\text{poly(L)}}$. Further by claim 4, $\frac{vol(E_i)}{vol(E_0)} \le \left(1 - \frac{1}{3n}\right)^i$. Therefore,

$$\begin{aligned} i &\le \frac{\log(vol(E_i)/vol(E_0))}{\log\left(1 - \frac{1}{3n}\right)} \\ &\le O(n) \cdot \log(2^{\text{poly}(L)}/2^{-\text{poly}(L)}) \\ &= O(n) \cdot \text{poly}(L) \\ &= \text{poly}(L). \end{aligned}$$

With above argument, we will get this algorithm terminates within poly(L) iterations. □

Reference

a) `http://www.cs.cmu.edu/~odonnell/toolkit13/lecture13.pdf`

b) `https://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15859-f11/www/notes/lecture08.pdf`