CSC2411 - Linear Programming and Combinatorial Optimization*
Lecture 6: The Ellipsoid Method: an Introduction

Notes taken by Svetlana Stolpner

March 6, 2007

Summary: This lecture introduces the Ellipsoid Method, the first polynomial-time algorithm to solve LP. We start by discussing the historical significance of its discovery by L. Khachiyan. Next, we argue that the ability to decide the feasibility of a version of the constraints of an LP is as hard as solving the LP. To gather the intuition behind the Ellipsoid Method, we draw an analogy with a problem of finding a large creature in a constrained space. Through this analogy, we formulate 3 "ingredients" that we require of the potential algorithm that will find this creature. Next, we show that each of these ingredients exists for the feasibility problem we are solving. Finally, we discuss the need for ellipsoids as the bounding volumes used to locate the feasible set. As a result, we are able to sketch the pseudocode of the Ellipsoid Method, whose correctness and polynomial time complexity is demonstrated. As preparation for the upcoming lecture, we consider expressing an ellipsoid using a positive semi-definite matrix.

1 Historical Background

The only algorithm for solving linear programming (LP) problems we have seen thus far is the Simplex Algorithm. We have also shown that this algorithm has exponential running time for certain rather contrived examples (due to Klee and Minty, 1972). We know that LP is in the complexity class $\mathcal{NP}$ (a solution x can easily be checked to satisfy a corresponding decision problem $Ax = b$, $x \geq 0$, $\langle x, c \rangle \leq \lambda$ in polynomial time). Moreover, LP is in the class co-$\mathcal{NP}$ (as the solution to the dual decision problem can also be verified in polynomial time). It is conjectured that $\mathcal{P} = \mathcal{NP} \cup$ co-$\mathcal{NP}$, in other words, that problems such as LP have polynomial time algorithms. However, it was not until 1979 that such an algorithm was discovered. Due to the Soviet mathematician Leonid Khachiyan, the *ellipsoid method* is of an entirely different nature than the combinatorial simplex algorithm and has roots in nonlinear optimization. However,

* Lecture Notes for a course given by Avner Magen, Dept. of Computer Sciecne, University of Toronto.

for practical purposes, the simplex method remains superior to the ellipsoid method on average; in fact, the expected running time of the simplex method is polynomial and faster than the expected running time of the ellipsoid method [1].

2 Starting Point

As we will see, the ellipsoid method is only *weakly* polynomial, as the number of operations of the algorithm is polynomial in the size, L, of the problem, where $L = n \cdot m + \lceil \log |P| \rceil$, and P is the product of the non-zero coefficients in A, b and c [1].

Consider the LP in standard form:

$$\min\langle c, x\rangle$$

$$Ax = b, x \geq 0.$$

Instead of solving this program, we are simply going to determine the feasibility of a program of linear inequalities (LI),

$$Ax \leq b.$$

If we were able to determine the feasibility of $Ax \leq b$, we could determine the feasibility of $Ax \geq b, Ax \leq b, x \geq 0$, i.e. check that the LP is feasible. Next we would check that the objective function is bounded by checking if $\langle c, x\rangle \leq -2^{2L} - 1^2$. Now that we know that the LP is feasible and bounded, we can perform *binary search* on the values of a in $Ax \geq b, Ax \leq b, x \geq 0, c'x \leq a$ to look for the optimal BFS $\hat{x}$ in time proportional to L. The search uses the lower bound on the difference between 2 BFSs to decide when to stop. (Actually, we find the basis that corresponds to $\hat{x}$ and then invert the corresponding matrix to find $\hat{x}$.) For a more detailed discussion, please see [3]. Therefore, a polynomial-time solution to LI gives a polynomial-time algorithm for LP. This also means that the complexity of the initialization stage of the Simplex method is same as that of the method in total.

We are still not quite ready for the ellipsoid method. Instead of determining the feasibility of $Ax \leq b$, we are going to determine the feasibility of $A'x < b'$ (with strict inequality in every coordinate) for some A', b' possibly different from A and b.

Note that taking $A = A'$ and $b = b'$ doesn't usually work. Consider for example $A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 1 \end{pmatrix}$ and $b = \begin{pmatrix} 2 \\ 3 \\ -5 \end{pmatrix}$. In this example, the only possible assignment of x is one that results in an equality. Therefore, we need to add some slack to the b's, i.e. consider

$$Ax < b + \epsilon = b + \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix} = b'.$$

[1]It may be necessary to normalize the coefficients first to be integers. The effect of this operations is easily reversed once the solution is available.

[2]Note that this is the answer to question 1 of assignment 2.

Lemma 2.1. *$Ax \leq b$ is feasible iff $Ax < b'$ is feasible.*

Proof. $(\Rightarrow)$ This direction clearly follows.
$(\Leftarrow)$ Consider the contrapositive, i.e. show that $Ax \leq b$ is not feasible $\Rightarrow Ax < b'$ is not feasible.

Applying Farkas' Lemma, the above statement is equivalent to:

$$\exists y \geq 0 \text{ such that } yA = 0 \text{ and } \langle y, b\rangle < 0 \Rightarrow \langle y, b'\rangle < 0.$$

$\langle y, b\rangle < 0$ may be scaled positively to become $\langle y, b\rangle = -1$. We verify this statement by checking that the y that satisfies the left-hand side also satisfies the right-hand side. Let $\overrightarrow{\mathbf{1}} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$.

$$\begin{aligned} \langle y, b'\rangle &= \langle y, b\rangle + \epsilon\langle y, \overrightarrow{\mathbf{1}}\rangle \\ &= -1 + \epsilon\langle y, \overrightarrow{\mathbf{1}}\rangle \\ &\leq -1 + \epsilon n 2^{2L}, \end{aligned}$$

where the last inequality follows because y is a BFS and in the worst case all the coordinates of y meet the maximum. By taking $\epsilon < \frac{1}{n2^{2L}}$, we get $\langle y, b'\rangle < 0$, as desired. $\square$

Therefore, we can conclude that to solve a general LP, it is enough to answer the question, "is $Ax < b$ feasible?".

3 Detour through paleontology

Suppose you are a daring paleontologist, looking to challenge your colleagues' theory that mammoths did not inhibit Antarctica. You travel to the South Pole and check whether a frozen mammoth is present there. If it is, you are done. If it is not, you are supplied with a dividing line through the South Pole and a guarantee that on one side of this line there is no mammoth [3]. You continue the search in the remaining part of Antarctica, first checking a location in this remaining territory and, if unsuccessful, dividing the remaining space with a line and as before, using the knowledge that a mammoth does not lie to one side of this line to decrease your search space. You continue this process until you find the mammoth, or until you reach a point when the space you are left with could not fit such a large creature. Refer to Figure 1 for an illustration.

Observe that the following "ingredients" are necessary for the method to work:

1. The initial search space (i.e. Antarctica) is relatively small

2. You have a lower bound on the size of the desired object (the mammoth)

[3] Suppose that a "guiding spirit", or an "oracle" is supplying this information

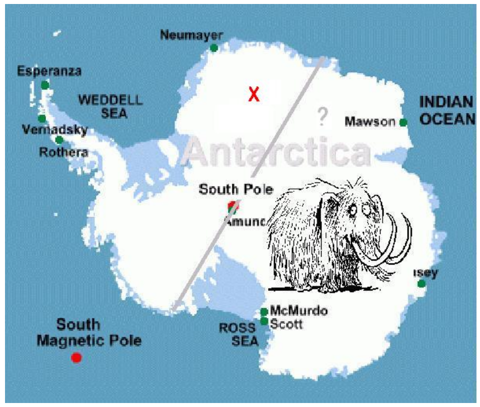

Figure 1: The problem of finding a mammoth in Antarctica.

3. You have a way to pick the next search space that will have the following properties

   - It always contains the desired object (mammoth), if it exists.
   - Its area is considerably smaller than the area of the previous search space.
   - The search space can be easily represented and maintained.

4 The intricate connection between finding mammoths and locating feasible sets

We are interested to know whether the set $\{x|Ax < b\}$ is non-empty. We know that if it is, it must exist in a reasonably limited area. Namely, if $Ax < b$ is feasible and bounded, $|x_i| < 2^{2L} \forall i$. In other words, the feasible region exists inside a hypercube centered at the origin with side lengths 2×2^{2L}. In case $\{x|Ax < b\}$ is unbounded, the above hypercube would contain any BFS of the problem, if they exist. For a formal discussion of the issue of an unbounded feasible set, refer to page 33 of [3]. Also refer to Figure 2 for clarification. Therefore, we have a bound on the initial search space (ingredient 1). In fact, instead of considering the cube discussed above, we are going to be working with the ball $B = \{\|x\| \leq 2^{2L}\ \}$ centered at the origin that is enclosed in this cube. If $\{x|Ax < b\} \neq \emptyset$, $\{x|Ax < b\} \bigcap B \neq \emptyset$.

Now we seek a lower bound on the volume of $P = \{x|Ax < b, |x_i| < 2^{2L} \forall i\}$ (ingredient 2). Since P is an intersection of strict inequalities, it is an open set (i.e. around any point in P there is an ϵ-ball in P) and so, it is of full dimension. Hence vol$(P) > 0 \Leftrightarrow P \neq \emptyset$. However, we are looking for a more useful lower bound on vol(P).

Lemma 4.1. $P \neq \emptyset \Rightarrow$ *vol*$(P) > 2^{-n^2-2nL}$

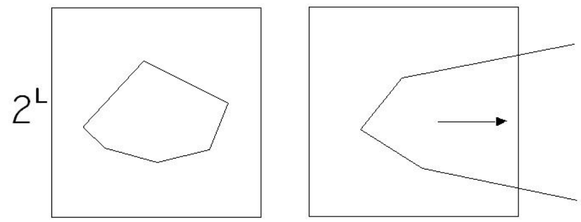

Figure 2: The feasible region is contained inside a cube. In case it is unbounded (right), a part of this region is still contained in the cube. Image adopted from [4].

Proof. Instead of P, let's consider the closure of P, $\bar{P}$. This is valid since vol$(\bar{P}) =$ vol(P). As P is of full dimension in $\mathbb{R}^n$, if P is non-empty, there are $n+1$ affinely independent BFSs in P[4], $S = \{v_0, \ldots, v_n\}$. By the convexity of P and using the formula for the volume of a simplex with vertices S,

$$\text{vol}(P) = \text{vol}(\bar{P}) \geq \text{vol}(\text{conv}(S)),$$

$$\text{vol}(\text{conv}(S)) = \frac{1}{n!}\left|\det\begin{pmatrix} 1 & \ldots & 1 \\ v_0 & \ldots & v_n \end{pmatrix}\right|.$$

As we are assuming that the volume is non-zero, the determinant is non-zero and the volume can be lower-bounded using the inverse of the absolute value of the maximum denominator achievable, 2^{-2L}, as follows:

$$\begin{aligned} \text{vol}(\text{conv}(S)) &> \frac{1}{n! 2^{n2L}} \\ &\geq 2^{-n^2-2nL}, \end{aligned}$$

using the fact that $n! \leq 2^{n^2}$ and since the determinant of a matrix is a sum of terms that are each a product of $n-1$ terms having denominators at most 2^{2L}

since we showed that the denominators of v_i are upper bounded by 2^L in assignment 1, question 3, and using the bound on $n!$ also shown in the solution to this question. $\square$

Now we need to satisfy ingredient 3. We start by checking the centre x_0 of the ball B. If x_0 is feasible, we know that P is feasible and stop. If x_0 is not feasible, we know there is at least one inequality $\langle a_i, x\rangle < b'_i$ that is violated at x_0, i.e. $\langle a_i, x_0\rangle \geq b'_i$, and it is easy to find. So, this means that we can restrict our search space to $\langle a_i, x\rangle\rangle b'_i$. Formally, if we know that our search space $E \subset P = \{x|Ax \leq b\}$, then $(\{x|\langle a_i, x\rangle < b_i\} \bigcap E) \supset P$, where i is the index of the inequality that is violated. This gives us the considerable reduction in search space required.

[4]This means that these $n+1$ points generate an affine space of dimension n in P. They must exist, because as P is the convex hull of its BFSs, if there are fewer than $n+1$ affinely independent BFSs, P would not be of full dimension n.

5 Towards ellipsoids

If the original region E is a sphere, then $\{x|\langle a_i, x\rangle < b_i\} \bigcap E$ is a half-sphere. If we insist on using only spheres as our bounding body, then the sphere that bounds $\{x|\langle a_i, x\rangle < b_i\} \bigcap E$ is E and we get no reduction in search space size. We must move toward a bounding body that has more flexibility.

Observation 5.1. *If E is an ellipsoid and $\frac{1}{2}E$ is a "half-ellipsoid"* [5]*, then $\exists E' \supset \frac{1}{2}E$ so that $vol(E') \leq e^{-\frac{1}{2n}}vol(E)$.*

Proof. The proof of this observation in [2] is technical and lengthly. [3] shows a slightly different bound, but with a more elegant presentation. □

6 The algorithm sketch

Algorithm 6.1 (The Ellipsoid Method).

Input: A, b, Volume Lower Bound (VLB) = 2^{-2nL-n^2}, radius of the initial ball $R = 2^{2L}$, $E_0 = B(0, R)$, the ball centered at 0 with radius R.

Output: A declaration on the feasibility of $\{x|\langle a_i, x\rangle < b_i \forall i\}$

$E = E_0$.

while vol$(E) > VLB$,

if $y =$ centre(E) is feasible by each of the m inequalities.

Declare "feasible". Stop.

else Find inequality i such that $\langle a_i, y\rangle \geq b_i$.

Let E' be an ellipsoid that contains $E \bigcap \{x|\langle a_i, x\rangle \leq b_i\}$ and vol$(E') < e^{-\frac{1}{2n}}$vol(E).

$E = E'$.

Declare "not feasible". Stop.

Refer to Figure 3 for an intuitive illustration of the algorithm.

6.1 Correctness

If the algorithm declares "feasible", we have a feasible point in P, y. If the algorithm declares "not feasible", it is because vol$(E) < VLB$, and we have already shown in Lemma 4.1 that if P is feasible, then its volume is lower-bounded by 2^{-2nL-n^2}.

[5] A half-ellipsoid is the intersection of an ellipsoid with a half-space through its centre

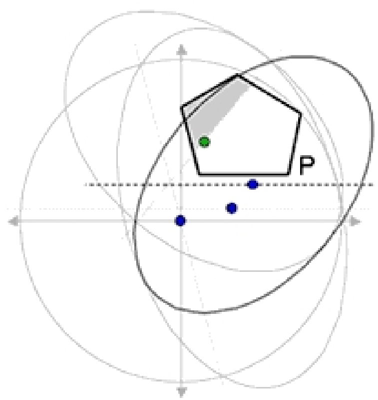

Figure 3: The ellipsoid method converging on the feasible region. The blue points are centres of ellipsoids considered that are not inside the feasible region. The last ellipsoid considered is centered at the green point, which is inside the feasible region. The dashed line corresponds to the unsatisfied inequality for the centre of the next-to-last ellipsoid. Image from [5].

6.2 Running Time

The volume of E_0 is upper-bounded by the volume of the n-dimensional cube having side lengths 2^{2L+1}. We also know that the final volume E is lower-bounded by VLB. Therefore,

$$\begin{aligned} \text{initial volume} &< 2^{(2L+1)n} \\ \text{final volume} &\geq 2^{-2nL-n^2} \end{aligned}$$

If T iterations are required to determine the feasibility of P,

$$\text{final volume} \times (e^{\frac{1}{2n}})^T \leq \text{initial volume}.$$

Now we solve for T:

$$\begin{aligned} T &= \log_{e^{\frac{1}{2n}}} \left(\frac{\text{initial volume}}{\text{final volume}} \right) \\ &\simeq \frac{\log \frac{2^{(2L+1)n}}{2^{-2nL-n^2}}}{\log e^{\frac{1}{2n}}} \\ &= O(nL+n^2)2n \\ &= O(n^2L+n^3) \end{aligned}$$

Therefore, the ellipsoid algorithm performs a polynomial number of iterations as a function of n.

7 Ellipsoids and semi-definite matrices

Consider the unit ball at the origin: $B(0,1) = \{x | \|x\| \leq 1\}$. Let T be an affine transformation $T : x \rightarrow Ax + z$. The image of $B(0,1)$ under T is an ellipsoid $E = \{y | y = Ax + z, \|x\| \leq 1\}$. Since $x = A^{-1}(y - z)$,

$$
\begin{aligned}
y \in E \quad &\Leftrightarrow \quad \|A^{-1}(y-z)\| \leq 1 \\
&\Leftrightarrow \quad (y-z)^t (A^{-1})^t A^{-1} (y-z) \leq 1 \\
&\Leftrightarrow \quad (y-z)^t Q^{-1} (y-z) \leq 1,
\end{aligned}
$$

where $Q = AA^t$, a positive, semi-definite matrix. If $z = \text{centre}(E)$ and $Q = I$, then $E = B(z,1)$. If $Q = \sqrt{r} I$, then $E = B(z,r)$.

References

[1] K-H. Borgwardt. (1982), "Some distribution-independent results about the asymptotic order of the average number of pivot steps of the simplex method," *Mathematics of Operations Research*, 1982, pp. 441-462.

[2] M. Grötschel, L. Lovász, A. Schrijver. Geometric Algorithms and Combinatorial Optimization, 2nd Corrected Edition, `Springer-Verlag`, 1993, pp. 64-86.

[3] C. H. Papadimitrou, K. Steiglitz. Combinatorial Optimization: Algorithms and Complexity, `Dover`, 1998, pp. 170-185.

[4] X. He. Lecture 7: "von Neumann minimax theorem, Yao's minimax Principle, Ellipsoid Algorithm," *Lecture Notes for CSC2411*, 2005.

[5] `http://www.orie.cornell.edu/orie/academics/phd/fields.cfm`