Mathematical Tools in Computer Science, Winter 2006
Lecture 8
Lecturer: Nati Linial — *Scribe: Ran Berenfeld*

1 Linear Programming and The Ellipsoid Method

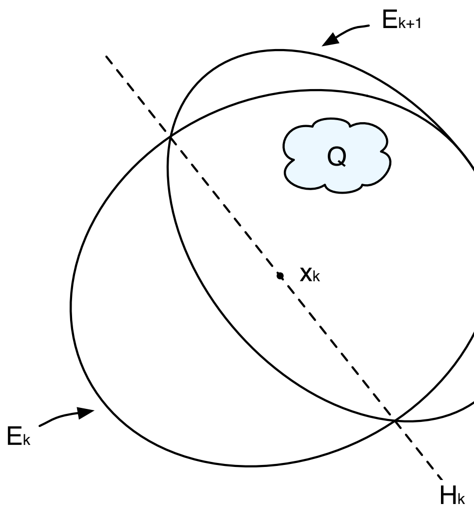

Figure 1: A graphical depiction of one iteration of the ellipsoid algorithm.

Every linear programming problem

$$\text{maximize } \langle \mathbf{x}, \mathbf{c} \rangle \quad \text{s.t.} \quad A\mathbf{x} \leq \mathbf{b},\ \mathbf{x} \geq 0 \ \ ,$$

can be reduced to a feasibility problem, namely, the problem of determining whether a convex set P is empty or not. Specifically, given the linear program dual

$$\text{minimize } \langle \mathbf{y}, \mathbf{b} \rangle \quad \text{s.t.} \quad \mathbf{y}^\top A \geq \mathbf{c},\ \mathbf{y} \geq 0 \ \ ,$$

we search for a point in the convex set

$$P = \{(\mathbf{x}, \mathbf{y}) : \ A\mathbf{x} \leq \mathbf{b},\ \mathbf{y}^\top A \geq \mathbf{c},\ \langle \mathbf{x}, \mathbf{c} \rangle = \langle \mathbf{y}, \mathbf{b} \rangle,\ \mathbf{x} \geq 0,\ \mathbf{y} \geq 0\}$$

How is the convex body P represented ? Using two oracles: a *membership oracle* and a *separation oracle*. The membership oracle can answer the question, "is $\mathbf{x}$ in P", for any $\mathbf{x} \in \mathbb{R}^n$. The separation oracle takes a vector $\mathbf{x} \notin P$ and returns a hyperplane which separates P from $\{\mathbf{x}\}$.

If P is a bounded set, we can solve the convex feasibility problem using the *ellipsoid method*. Assume that we know parameters R and r, such that P is contained within a ball of radius R about the origin, and if $P \neq \emptyset$ then it contains a ball of radius r. The ellipsoid algorithm iteratively constructs a sequence of ellipsoids, $E_1, E_2, \ldots$, such that $\mathrm{Vol}(E_k) > \mathrm{Vol}(E_{k+1})$ for all k and $P \subseteq E_k$ for all k. If the center of one of these ellipsoids is contained in P then $P \neq \emptyset$ and the algorithm stops. Otherwise, the algorithm eventually constructs an ellipsoid whose volume is less than that of a ball of radius r. When this happens, we conclude that $P = \emptyset$. The running time of the algorithm is therefore proportional to the rate at which the volume of the ellipsoids decreases.

More concretely, let E_k denote the k'th ellipsoid in the sequence, constructed on iteration $k-1$. Let $\mathbf{x}_k$ denote the center of this ellipsoid. Iteration k begins by using the membership oracle to determine whether $\mathbf{x}_k \in P$. If the answer is yes, then P is non-empty and the algorithm stops. Otherwise, the algorithm uses the separation oracle to find a hyperplane H_k which separates P from the set $\{\mathbf{x}_k\}$. We can assume w.l.o.g. that this separating hyperplane passes through $\mathbf{x}$, and therefore splits the ellipsoid into two equal halves. Let H_k^+ denote the half-space defined by this hyperplane which contains P. The algorithm now defines E_{k+1} to be the smallest ellipsoid which contains $E_k \cap H_k^+$.

Next, we will derive the explicit recursive definition of E_k, and prove that $\mathrm{Vol}(E_{k+1}) \leq \mathrm{Vol}(E_k)(1 - \frac{1}{2n})$. Finally, we will show why this rate of volume decrease is fast enough for the ellipsoid algorithm to run in polytime.

Lemma 1 *Given a convex set P and a membership and separation oracle then $Vol(E_{k+1}) \leq$ $\mathrm{Vol}(E_k)(1 - \frac{1}{2n})$.*

Proof We will prove it for the case where E_k is the unit ball $B(0,1)$ and the separating hyperplane $x_1 = 0$. Let $H_k^+ = \{\mathbf{x} \in \mathbb{R}^n : \ x_1 > 0\}$ we find the smallest ellipsoid E_{k+1} containing $E_k \cap H_k^+$. We move the ellipsoid center E_{k+1} towards the positive ray of x_1 by considering $\mathbf{u} = t\mathbf{e}_1$ and $E_{k+1} = \{\mathbf{x} \in \mathbb{R}^n : \ (\mathbf{x} - \mathbf{u})^\top M_{k+1}(\mathbf{x} - \mathbf{u}) = 1\}$, where M_{k+1} is a positive definite diagonal matrix with $M_{k+1}(1,1) = \alpha$ and $M_{k+1}(i,i) = \beta$ for $i = 2, ..., n$.

- $(1, 0, ..., 0) \in E_{k+1}$ translates to the requirement $(1-t)^2\alpha = 1$.
- $(0, t_2, ..., t_n) \in E_{k+1}$ for $\sum_{i=2}^n t_i^2 = 1$ translates to the requirement $\alpha t^2 + \beta = 1$.

We wish to find the minimal $Vol(E_{k+1}) = \frac{Vol(E_k)}{sqrt(\det M_{k+1})}$. This problem can be solved analytically and the minimum is attained $t = \frac{1}{n+1}$: By the above facts $\alpha = \frac{1}{(1-t)^2}$ and $\beta = \frac{1-2t}{(1-t)^2}$. To find the minimal volume we derive: $\frac{d}{dt}\alpha\beta^{n-1} = 0 \iff t = \frac{1}{n+1}$.

To estimate $\alpha\beta^{n-1} = \frac{n+1}{n-1}\left(1 - \frac{1}{n^2}\right)^n$ we estimate $\frac{n+1}{n-1} = 1 + \frac{2}{n} + o(\frac{1}{n})$ and $\left(1 - \frac{1}{n^2}\right)^n = 1 - \frac{1}{n} + o(\frac{!}{n^2}$. We conclude that the volume of E_{k+1} is less than $\left(1 - \frac{1}{2n}\right) Vol(E_k)$. ■

We turn to prove that the ellipsoid algorithm solves the feasibility problem in polynomial time in the size of the input:

Theorem 2 *Let P be a convex set containing a ball of radius r and contained in a ball of radius R, such that $\frac{R}{r} \leq e^{poly(n)}$. If there are membership and separation oracle for P then the ellipsoid algorithm decides in time $poly(n)$ whether $P = \emptyset$.*

Proof The volume of E_0 is at most as the volume of a ball of radius R, and for every k the volume of E_k is at least as the volume of a ball of radius r. After performing k iterations $vol(E_k) \leq vol(E_0)(1 - \frac{1}{2n})^k$, thus after $T \geq \frac{n \log(R/r)}{\log(1+\frac{1}{2n-1})}$ iterations the volume of E_T is less than the volume of a ball of radius r, therefore we conclude that the set P is empty. A Taylor series expansion of $\log(1 + \frac{1}{2n}) = \frac{1}{2n} + o(\frac{1}{n^2})$ implies that $T = \text{poly}(n)$. ■