Optimization	**Summer 2016**

Lecture 15: June 13, 2016

Lecturer: Parinya Chalermsook Scribe: Marius Mosbach, Jannik Luxenburger

1 Recap

The simplex algorithm – which we studied in the first half of this class – is a prominent algorithm proposed for linear programming and has shown to be very fast in practice, however, there is yet no variant of the simplex algorithm which is shown to run in polynomial time in general. On the other hand, the Ellipsoid algorithm – which was first applied to linear programming problems by Leonid Khachiyan in 1979 – is the first polynomial time algorithm for linear programming. Besides being efficient in theory, the ellipsoid algorithm is not very fast in practice (other than simplex), however, it plays an important role in theory.

We start today's lecture by recapping the concept of a *separation oracle* and by stating the *Ellipsoid Theorem.*

1.1 Separation Oracle and the Ellipsoid Theorem

Definition 1. ***Separation oracle***
Given a polytope $P \subseteq \mathbb{R}^n$ and a point $x^* \in \mathbb{R}^n$, a separation oracle either decides that $x^* \in P$ or finds an inequality $a^T x \leq b$ which is satisfied by all $x \in P$ but $a^T x^* > b$.

Theorem 1. ***Ellipsoid Theorem***
Given a polytope $P \subseteq \mathbb{R}^n$. We can find a point $x \in P$ by calling the separation oracle poly(n)-times.

(Notice that if we have an efficient algorithm which finds a feasible point in P, then there also exists an efficient algorithm that computes $max\{c^T x \mid x \in P\}$. For the remainder of this lecture we assume this to hold. A proof will be given in the tutorial.)

2 Background

2.1 Linear Algebra Background

To get started with our survey on the ellipsoid method we first need to establish some linear algebra background.

Definition 2. *Let A be a symmetric matrix, i.e $A^T = A$. λ is said to be an* ***eigenvalue*** *of A if $\exists\ x \neq 0$ s.t. $Ax = \lambda x$.*

Theorem 2. *Suppose $\lambda_1, \ldots, \lambda_n$ are eigenvalues of matrix A, then:*

1. *$det(A) = \prod_i \lambda_i$*
2. *$tr(A) = \sum_i \lambda_i$*

Definition 3. *A matrix A is said to be* ***positive semi-definite*** *if $\forall x.\ x^T A x \geq 0$.*

Theorem 3. *Let A be an arbitrary matrix. The following are equivalent:*

1. *A is positive semi-definite.*
2. *All eigenvalues of A are non-negative.*
3. *$\exists B$ s.t. $A = B^T B$. ($\rightsquigarrow B = A^{\frac{1}{2}}$)*

Definition 4. ***Ellipsoid***
Let A be a positive semi-definite matrix. The ellipsoid $E(A, a)$ is defined as follows:

$$E(A, a) = \{x \mid (x - a)^T A^{-1} (x - a) \leq 1\},$$

where a is the center of the ellipsoid.

3 The Ellipsoid Algorithm

We start the discussion on the ellipsoid algorithm by giving a high level overview of how it works.

Given a polytope $P \subseteq \mathbb{R}^n$. We start with an initial ellipsoid E_0 that contains P. In each iteration k we check if the center of the the ellipsoid E_k is contained in P. If so, we are done. If not, we use the separation oracle to find a separating hyperplane H that separates the center of the ellipsoid E_k from P. Having found such a separating hyperplane we construct a new ellipsoid E_{k+1} of minimal volume which contains the set $E_k \cap H$ and start the next iteration.

The abstract algorithm looks as follows:

1. $E_0 =$ initial ellipsoid containing P
2. **While** center a_k of E_k is not in P do:

- Find a separating hyperplane $c^T x$ s.t. $P \subseteq \{x \mid c^T x \leq c^T a_k\}$
- Construct minimum-volume ellipsoid E_{k+1} which contains $E_k \cap \{x \mid c^T x \leq c^T a_k\}$
- $k = k + 1$

To show that the algorithm above terminates we need to proof the following theorem.

Theorem 4. ***Volume Reduction***

$$\frac{vol(E_{k+1})}{vol(E_k)} \leq e^{-\frac{1}{2(n+1)}} < 1$$

Intuitively the theorem says that the next ellipsoid E_{k+1} is guaranteed to be smaller then its predecessor E_k.

Corollary. *The number of iterations of the ellipsoid algorithm is at most* $\mathcal{O}(n^2 * lg(\frac{R}{r}))$.

We start with a proof of the corollary but first we need some assumptions.

1. We assume that the volume reduction theorem holds.
2. The polyhedron P has non-zero volume, i.e. $\exists r > 0, c \in \mathbb{R}^n$ s.t. $Ball(c, r) \subseteq P$.

Proof. The volume of the n-dimensional sphere is given by:

$$vol(E_0) = \frac{\pi^{\frac{n}{2}} R^n}{\Gamma(\frac{n}{2}+1)}$$

Using assumption (2) from above we get the following invariant:

$$\forall i.\ P \subseteq E_i.\ \text{Hence, } vol(E_i) \geq vol(Ball(c, r) \geq \frac{\pi^{\frac{n}{2}} r^n}{\Gamma(\frac{n}{2}+1)}$$

Using assumption (1) we get:

$$(\frac{r}{R})^n = \frac{vol(E_k}{vol(E_0)} = \frac{vol(E_k}{vol(E_{k-1})} * \frac{vol(E_{k-1}}{vol(E_{k-2})} * \cdots * \frac{vol(E_1}{vol(E_0)} \leq e^{-\frac{k}{2(n+1)}}$$

This gives us:

$$e^{\frac{k}{2(n+1)}} \leq (\frac{R}{r})^n$$

Taking the logarithm on both sides we get:

$$\frac{k}{2(n+1)} \leq n * lg((\frac{R}{r}))$$

Hence,

$$k \leq 2n * (n+1) * lg((\frac{R}{r}))$$

Which proofs the claim.

$\square$

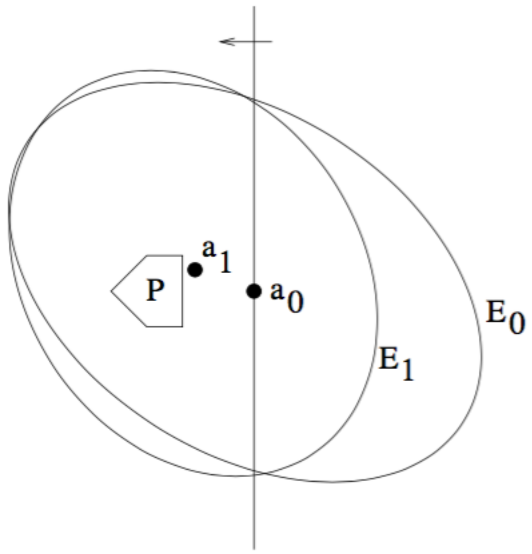

Figure 1: A single iteration of the ellipsoid method.