6.966 Algebra and Computation	December 2, 1998
Lecture 24	
Lecturer: Madhu Sudan	*Scribe: Sofya Raskhodnikova*

In this lecture we investigate topological methods for obtaining lower bounds on the depth of real computation trees which decide the membership problem for a subset Π of $\mathbb{R}$. We study the theorem due to Ben-Or which gives a lower bound in terms of the number of connected components of Π and apply it to several problems. We also discuss lower bounds in terms of volume and mention a generalization of Ben-Or's technique to Euler characteristic and Betti numbers.

1 Connected Components

1.1 Model

All computation trees we consider in this lecture are defined over the reals and allow both arithmetic operations and tests for equality and inequality to zero. The lower bound technique we are going to study is so powerful that we will count only the operations from $\{\times, \div, \sqrt{}, = 0, \geq 0\}$. Each of these operations can be done for a unit cost. Addition, subtraction, and multiplication by scalars from $\mathbb{R}$ are performed for free.

1.2 Motivating Example and Statement of the Ben-Or's Theorem

Example 1 [Element Distinctness Problem] *Given n real numbers, determine if two of them are equal. Equivalently and more formally,*

$$\mathrm{ED}_n = \{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid \prod_{i \neq j}(x_i - x_j) = 0\}.$$

Element Distinctness can be solved in $O(n \log n)$ time given ≥ 0 and $-$ operations. In comparison-based model it is not hard to show that Element Distinctness requires $\Omega(n \log n)$ comparisons. However, comparison-based model does not allow serious computation. We will investigate whether a less costly solution to this problem is feasible in our model.

Steel and Yao proved a lower bound of $\Omega(n \log n)$ for the case when only "linear" operations are allowed. Today we will study a more general result by Ben-Or which uses topological invariants to obtain a lower bound on the complexity of real computation trees.

Theorem 1 [Ben-Or] *If* Yes *instances of some problem Π have N distinct connected components in $\mathbb{R}^n$, then the depth of the real computation tree for this problem is $\Omega(\log N - n)$.*

Applying Theorem 1 to the complement of Π, further denoted as $\overline{\Pi}$, we get the same statement for No instances.

Let us give an application of this result before we attempt to prove it.

Theorem 2 *Any algebraic computation tree solving Element Distinctness problem for n numbers must have depth of $\Omega(n \log n)$.*

Proof Yes instances of this problem (the points in ED_n) form one connected component in $\mathbb{R}^n$ because ED_n contains a continuous path from any point with two equal coordinates to the origin. Therefore, in this case, applying Theorem 1 to the Yes instances of the problem is not particularly useful for obtaining a lower bound.

However, the number of connected components, N, for No instances of Element Distinctness (the points in $\overline{\mathrm{ED}_n}$) is $n!$ because we have one component for every permutation $\pi : [n] \rightarrow [n]$. To see this, observe that any No instance $(x_1, \ldots, x_n)$ satisfying condition $x_{\pi(1)} < x_{\pi(2)} < \cdots < x_{\pi(n)}$ is connected only to No instances for which this condition still holds. Applying Theorem 1, we get a lower bound of $\Omega(n \log n)$. ■

1.3 Bounding the Number of Connected Components of a Semi-Algebraic Set

Definition 3 *A subset S of $\mathbb{R}^N$ is called* **semi-algebraic** *if it is the set of all real solutions to a system of finitely many polynomial equalities and inequalities or a finite union of such sets.*

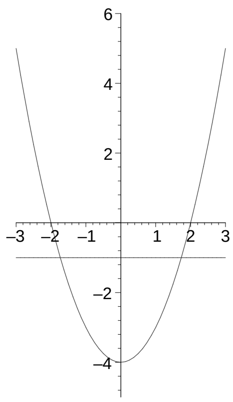
We are interested in bounding the number of connected components of a semi-algebraic set of the form

$$S = \left\{ \overline{x} \in \mathbb{R}^M \;\middle|\; \begin{array}{l} p_1(\overline{x}) = 0, \ldots, p_l(\overline{x}) = 0, \\ q_1(\overline{x}) \geq 0, \ldots, q_{K_1}(\overline{x}) \geq 0, \\ r_{K_1+1}(\overline{x}) < 0, \ldots, r_K(\overline{x}) < 0 \end{array} \right\} \tag{1}$$

To see that the number of connected components in the set S of this form can be more than one, consider the following example.

Example 2 *Look at the set of points in $\mathbb{R}^2$ defined by*

$$\begin{aligned} y - x^2 + 4 &\leq 0 \\ y + 1 &\geq 0 \end{aligned}$$



There are two distinct connected components. Both lie above the line $y = -1$. One is positioned to the left of the parabola $y = x^2 - 4$, the other is to the right of the parabola.

To derive an upper bound on the number of connected components of a semi-algebraic set given by a system of the form (1), we use without proof the following very powerful theorem.

Theorem 4 [Milnor-Thom] *(Real varieties).*
Let V be a variety in $\mathbb{R}^M$ defined by polynomial equalities $h_1 = 0, h_2 = 0, \ldots, h_L = 0$ of degree at most D. Then the number of connected components of V is at most $D(2D-1)^{M-1}$.

Notice that this statement resembles Bezout's Theorem. However, here we are not working over algebraically closed fields and instead of the number of irreducible components we are counting the number of connected components. This theorem is more powerful than Bezout's. Also observe that L, the number of polynomial equations, does not enter the bound.

As a corollary of Milnor-Thom theorem, we get the following result.

Theorem 5 *The number of connected components in S is at most $D(2D-1)^{M+K-1}$, where D is the maximum degree of polynomials defining S.*

Proof The main idea of the proof is to convert the inequalities defining the variety to equalities without decreasing the number of connected components, and then to apply the Minor-Thom Theorem.

Let r be the number of connected components. Pick points $v_1, \ldots, v_r$ from distinct connected components.

We start by converting strict inequalities to non-strict. Let $\epsilon = \max_{i,j}\{-r_i(v_j)\}$ where i ranges from $K_1 + 1$ to K and j is from 1 to r. Since $v_j \in S$, we know that $r_i(v_j) < 0$, and hence ϵ is negative. Replace $r_i < 0$ with $r_i \leq \epsilon$ or, equivalently, with $\epsilon - r_i \geq 0$. Set $q_i = \epsilon - r_i$.

Let S' be the variety defined by the modified set of polynomial equalities and inequalities. Then $v_1, \ldots, v_r \in S'$ and S' is contained in S. Since S' is a subset of S, all v_js are in distinct connected components of S'. Therefore,

$$\mathcal{CC}(S) \leq \mathcal{CC}(S'). \tag{2}$$

We are left with K non-strict inequalities of the form $q_i \geq 0$. We replace them with equalities of the form $q_i = z_i^2$ where z_i's are new variables. Let S'' be the variety in $\mathbb{R}^{M+K}$ defined by the modified set of polynomial equalities. Notice that S' is a projection of S'' onto the first n coordinates. By a simple topological fact[1],

$$\mathcal{CC}(S') \leq \mathcal{CC}(S''). \tag{3}$$

Finally, applying Milnor-Thor Theorem to $S'' \subseteq \mathbb{R}^{M+k}$ and combining the result with inequalities 2 and 3, we get

$$\mathcal{CC}(S) \leq \mathcal{CC}(S') \leq \mathcal{CC}(S'') \leq D(2D-1)^{M+K-1}.$$

■

1.4 Proof of the Ben-Or's Theorem

Now we are ready to derive the main theorem of the lecture.

Proof of Theorem 1 The idea of the proof is to look at the decision tree.

- Fix a decision tree T that decides Π. Let h be the depth of T.
- Fix an accepting leaf L in T.
- Consider set S of inputs in $\mathbb{R}^n$ whose path in T leads to L.
- **Bound the number of connected components in S.**
- Bound the number of connected components in Π based on the previous step.

The core of the proof is bounding the number of connected components in S.

Set S can be described by a system of equalities and inequalities. We assign indeterminates $x_1, \ldots, x_n$ to the inputs. Each node from the root to leaf L gets assigned a value v_i depending on the operation it performs. If it computes one of the free operations, its value is one of the following

$$\begin{aligned} v_i &= c \\ v_i &= v_j \pm v_k \\ v_i &= c \times v_j \end{aligned}$$

where c is a constant and v_i and v_j are values of the node's ancestors. If it computes $\times, \div$ or $\sqrt{}$, it is assigned an indeterminate y_i. Each node with a multiplication instruction $v_i = v_j \times v_k$ is associated with equation

[1] If $\mu : S' \rightarrow S''$ is a continuous surjective map of topological spaces, then $\mathcal{CC}(S') \leq \mathcal{CC}(S'')$.

$y_i - v_j * v_k = 0$, each node with a division instruction $v_i = v_j \div v_k$ is associated with equation $y_i v_k - v_j = 0$, and each node with a square root instruction $v_i = \sqrt{v_j}$ is associated with equation $y_i^2 - v_j = 0$.[2]

A branching node with a test instruction of the form $v_i \geq 0$ is assigned the inequality $v_i \geq 0$ ($v_i < 0$) if the path proceeds along the YES (NO) branch, respectively. Finally, to a branching node with a test instruction of the form $v_i = 0$? we assign the equality $v_i = 0$ if the path proceeds along the YES branch. If the path continues with the NO branch, we employ (guess what) Rabinovich's trick, i.e. introduce a new indeterminate y_i and assign the equality $y_i v_i - 1 = 0$ to the node.

Observe that all introduced equalities and inequalities have degree at most two.

$$D = 2$$

The number of inequalities, call it k, is equal to the number of ≥ 0 instructions along the path to L. The number of indeterminates y_i, call is m, is bounded above by the number of operations from $\{\times, \div, \sqrt{}, = 0\}$. Since each tree node introduces either at most one new indeterminate or at most one inequality, we know that

$$k + m \leq h,$$

where h is the depth of T. Now S can be defined using above equalities and inequalities as follows.

$S = \{x_1, \ldots, x_n \mid \exists y_1, \ldots y_m$ satisfying

$$\begin{gathered} p_1(\overline{x}, \overline{y}) = 0, \ldots, p_l(\overline{x}, \overline{y}) = 0, \\ q_1(\overline{x}, \overline{y}) \geq 0, \ldots, q_{k_1}(\overline{x}, \overline{y}) \geq 0, \\ r_{k_1+1}(\overline{x}, \overline{y}) < 0, \ldots, r_k(\overline{x}, \overline{y}) < 0\}. \end{gathered}$$

Let E be the set of solutions in $\mathbb{R}^{n+m}$ of the above equations and inequalities.

$E = \{x_1, \ldots, x_n, y_1, \ldots y_m$ satisfying

$$\begin{gathered} p_1(\overline{x}, \overline{y}) = 0, \ldots, p_l(\overline{x}, \overline{y}) = 0, \\ q_1(\overline{x}, \overline{y}) \geq 0, \ldots, q_{k_1}(\overline{x}, \overline{y}) \geq 0, \\ r_{k_1+1}(\overline{x}, \overline{y}) < 0, \ldots, r_{k_2}(\overline{x}, \overline{y}) < 0\}. \end{gathered}$$

Notice that S is a projection of E onto the first n coordinates. Hence, $\mathcal{CC}(V) \leq \mathcal{CC}(E)$. Applying Theorem 5 to E and substituting $D = 2$, we obtain

$$\mathcal{CC}(S) \leq \mathcal{CC}(E) \leq D(2D-1)^{M+K-1} = 2 \cdot 3^{n+m+k-1} \leq 2 \cdot 3^{n+h-1} \leq 3^{n+h}.$$

Since the number of accepting leaves in T is at most 2^h and each leaf has at most 3^{n+h} connected components, the total number of connected components is

$$N \leq 2^h 3^{n+h}.$$

Consequently,

$$h = \Omega(\log N - n).$$

■

[2] If we are interested in enhancing our model, we can easily add new operations. For example, the operation of taking the k-th square root can be handled as follows. The node with a k-th square root instruction $v_i = \sqrt[k]{v_j}$ would be associated with $O(\log k)$ equations of degree ≤ 2, each introducing a new variable s_i: $v_i = s_0, s_1 = s_0^2, s_2 = s_1^2, \ldots, v_j = s_{i'} \cdot s_{j'}$. The cost of this operation should be equal to the number of variables introduced, $O(\log k)$. We can add any operation of taking roots of a polynomial in a similar fashion.

1.5 Applications

1.5.1 Discriminant

Example 3 [Computing Discriminant] *Given real numbers $x_1, \ldots, x_n$, evaluate the discriminant polynomial $\prod_{i \neq j}(x_i - x_j)$.*

Theorem 6 *An algorithm for this problem requires $\Omega(n \log n)$ operations.*

Proof Observe that $(x_1, \ldots, x_n)$ is a YES instance of the Element Distinctness problem if and only if $\prod_{i \neq j}(x_i - x_j) = 0$. Therefore, our lower bound of $\Omega(n \log n)$ for the Element Distinctness problem implies the same lower bound for this problem. ■

1.5.2 The Convex Hull Problem

Let C be a convex set. A point $x \in C$ is an *extreme* point of C if there are no points y and z in C such that x lies on the open line segment between y and z.

Example 4 [The Convex Hull Problem in $\mathbb{R}^n$] *Given n points on the plane, determine if their convex hull has n extreme points, i.e. no point belongs to the convex hull of other $n-1$ points. Formally,*

$$\mathrm{CH}_n = \{(x_1, \ldots, x_n) \in (\mathbb{R}^2)^n \mid \textit{the convex hull of } x_1, \ldots, x_n \textit{ has } n \textit{ extreme points}\}.$$

There is an algorithm that decides this problem with $O(n \log n)$ operations in the worst case. We show that this algorithm is asymptoticly optimal.

Theorem 7 *Any computation tree solving the Convex Hull Problem has worse case complexity $\Omega(n \log n)$.*

Proof Consider $X \in \mathrm{CH}_n$. Let O denote the center of mass of $x_1, \ldots, x_n$. Since $x_1, \ldots, x_n$ form a convex set, O lies inside the region defined by $x_1, \ldots, x_n$. For i from 2 to n, let θ_i be the angle $x_1 O x_i$. For each of $(n-1)!$ permutations $\pi : [n-1] \rightarrow [n-1]$ of θ_is, there is a corresponding connected component in CH_n. To see this, observe that any X satisfying condition $\theta_{\pi(2)} < \theta_{\pi(3)} < \cdots < \theta_{\pi(n)}$ is connected only to YES instances for which this condition still holds. Applying Theorem 1, we get a lower bound of $\Omega(n \log n)$. ■

1.5.3 The Knapsack Problem

For some problems the Ben-Or's technique enables us to procure stronger lower bounds than $\Omega(n \log n)$.

Example 5 [The Knapsack Problem] *Given a set of n real numbers, determine if any its subset sums up to 1. Formally,*

$$\mathrm{KS}_n = \{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid \exists v \in \{0,1\}^n \textit{ such that } \sum_{i=1}^{n} x_i v_i = 1\}.$$

We know that Knapsack is **NP**-complete in Boolean world. Surprisingly, this problem can be solved with a computation tree of depth $O(n^5 \log^2 n)$ over the reals. We will prove a lower bound of $\Omega(n^2)$ for it.

Theorem 8 *The depth of an algebraic computation tree for the Knapsack problem has to be $\Omega(n^2)$.*

Proof The YES instances are connected. But luckily, the NO instances have a huge number of connected components. It is easy to see that two points $(x_1, \ldots, x_n)$ and $(y_1, \ldots, y_n)$ are in different connected components of $\overline{\mathrm{KS}_n}$ if there exists $v \in \{0,1\}^n$ such that

$$\sum_{i=1}^{n} x_i v_i > 1 \quad \text{and} \quad \sum_{i=1}^{n} y_i v_i < 1.$$

The remainder of the proof is devoted to obtaining a good lower bound on the $\mathcal{CC}(\overline{\mathrm{KS}_n})$.

Let B_n be the set of all Boolean functions $f: \{0,1\}^n \to \{0,1\}$. Each No instance naturally corresponds to a Boolean function f: if $\sum_{i=1}^n x_i v_i < 1$, then $f(v) = 0$; if $\sum_{i=1}^n x_i v_i > 1$, then $f(v) = 1$. For each f define

$$C_f = \{x \mid x \in \mathbb{R}^n \text{ and } \forall v \in \{0,1\}^n \ \sum_{i=1}^n x_i v_i < 1 \text{ if } f(v) = 0 \text{ and } \sum_{i=1}^n x_i v_i > 1 \text{ if } f(v) = 1\}. \tag{4}$$

Intuitively, C_f is the collection of No instances for which f is the corresponding Boolean function.

Notice that $\overline{\mathrm{KS}_n} = \bigcup_{f \in B_n} C_f$ and the C_fs are open and pairwise disjoint. Therefore, the number of nonempty C_fs is a lower bound on $\mathcal{CC}(\overline{KS_n})$. In other words, $\mathcal{CC}(\overline{KS_n}) \geq |G_n|$, where $G_n = \{f \mid f \in B_n$ and $C_f \neq \emptyset\}$.

Definition 9 *We say $f \in B_n$ is a* **threshold function** *if $\exists x_1, \ldots, x_n, \theta \in \mathbb{R}$ such that*

$$\forall v \in \{0,1\}^n \ \sum_{i=1}^n x_i v_i < \theta \text{ if } f(v) = 0 \text{ and } \sum_{i=1}^n x_i v_i > \theta \text{ if } f(v) = 1,$$

where θ is called a **threshold** *and $(x_1, \ldots, x_n)$ is a* **weight vector** *for f.*

Claim 10 *C_f is non-empty iff f is a threshold function with $f(\overline{0}) = 0$.*

Proof If $C_f \neq \emptyset$, then there exists a vector x satisfying (4). Therefore f is a threshold function with threshold 1 and weight vector x. Since $\sum_{i=1}^n x_i \cdot 0 = 0 < 1$, $f(\overline{0}) = 0$.

If f is a threshold function with $f(\overline{0}) = 0$, then the threshold must be greater than $\sum_{i=1}^n x_i \cdot 0 = 0$. Divide the weight vector by the threshold to obtain x that satisfies (4). ∎

Let T_n be the set of the threshold functions in B_n. The number of threshold functions f with $f(\overline{0}) = 1$ is equal to the number of threshold functions f with $f(\overline{0}) = 0$ because there is a one-to-one map between these two sets that takes $f(v)$ to $1 - f(v)$ (negate $x_1, \ldots, x_n, \theta$ to obtain a weight vector and a threshold for the new function). Consequently,

$$|T_n| = 2|G_n|. \tag{5}$$

Now our goal is to find a lower bound for $|T_n|$.

We will call a vector x *diverse* if all sums $\sum_{i=1}^n x_i v_i$ have different values for distinct v. Namely,

$$\forall v^{(1)} \neq v^{(2)} \in \{0,1\}^n \quad \sum_{i=1}^n x_i v_i^{(1)} \neq \sum_{i=1}^n x_i v_i^{(2)},$$

$$\text{or equivalently, for all non-zero } u \in \{-1,0,1\}^n \quad \sum_{i=1}^n x_i u_i \neq 0.$$

Given a weight vector for f we can wiggle its coordinates to obtain a diverse weight vector for f with the same threshold. By above argument $f \in T_n$ iff f has a diverse weight vector.

Claim 11 *If $x \in \mathbb{R}^n$ is a diverse vector, then it is a diverse weight vector for at least $2^n + 1$ threshold functions.*

Proof Since all sums $\sum_{i=1}^{n} x_i v_i$ have distinct values, they split $\mathbb{R}$ into $2^n + 1$ open intervals. Taking threshold θ from different intervals and setting $f(v) = 0$ iff $\sum_{i=1}^{n} x_i v_i < \theta$, we obtain $2^n + 1$ distinct threshold functions, as required. ■

Claim 12 $|T_n| \geq 2^{\binom{n}{2}+2}$.

Proof Consider the map $B_n \times B_n \longrightarrow B_{n+1}$ which takes $f, g \in B_n$ to

$$h(v_1, \ldots, v_{n+1}) = f(v_1, \ldots, v_n)v_{n+1} + g(v_1, \ldots, v_n)(1 - v_{n+1}).$$

Notice that $h \in T_{n+1}$ iff f and g have a common diverse weight vector. Hence, using Claim 11 we get

$$|T_{n+1}| \geq (2^n + 1)|T_n|.$$

Unfolding the recursion, we obtain

$$|T_n| \geq 2^{n-1} \cdot 2^{n-2} \cdot \cdots \cdot 2 \cdot |T_1| = 2^{\binom{n}{2}+2}.$$

■

Claim 12 and equation 5 give a lower bound of $2^{\binom{n}{2}+1}$ on the number of connected components of $\overline{KS_n}$. By Ben-Or's Theorem, the worst-case complexity of the Knapsack is $\Omega(n^2)$. ■

1.6 Limit of Connected Components Technique

We can prove a good lower bound for Knapsack using the Connected Components Technique, but it has a relatively low complexity in this model. Unfortunately, this technique is not strong enough for harder problems.

Suppose YES instances come from a semi-algebraic set S defined by degree D polynomials in $\mathbb{R}^N$. By Theorem 5 derived from the Milnor-Thom result, the number of connected components of S is at most $D(2D-1)^{N-1}$. Therefore, the best lower bound we can hope to prove using Ben-Or's theorem is $\Omega(N \log D)$. How large can D be for an $\mathbf{NP}_{\mathbb{R}}$-complete problem? D is at most exponential in the running time of the non-deterministic machine. Therefore the best lower bound we can prove is polynomial in the size of the input. This technique is too weak to prove $\mathbf{P}_{\mathbb{R}} \neq \mathbf{NP}_{\mathbb{R}}$.

2 Volume

2.1 k-Distinctness Problem

INPUT: $(x_1, \ldots, x_n) \in \mathbb{R}^n$.
YES INSTANCES: $\exists$ distinct $i_1, \ldots, i_k$ such that $x_{i_1} = x_{i_2} = \cdots = x_{i_k}$.

For $k = 2$ we already proved a lower bound of $\Omega(n \log n)$. However, as k increases, the problem becomes easier. In general, we can solve it with $O(n \log \frac{n}{k})$ comparisons.

Now both YES and NO instances form one connected component for $k \geq 2$. So the Connected Components method does not apply.

Bjorner, Lovasz and Yao '91 introduce a new technique which applies only to linear decision trees (each node can question if a linear function is greater or equal to zero). They look at volume of YES and NO instances to establish a lower bound. We will give an outline of their method.

- Fix a linear decision tree T that decides Π..

- Fix an accepting leaf L in T.
- The set of inputs in $\mathbb{R}^n$ whose path in T leads to L is convex because it is defined by linear equations and inequalities.
- Look at the largest convex set V defined by accepting leaves. Let U be an upper bound on the volume of V.
- Let L be a lower bound on the total volume of YES instances.
- The depth of the linear decision tree is lower bounded by $\log \frac{U}{L}$.

Using the Volume technique, we can obtain an optimal lower bound of $\Omega(n \log \frac{n}{k})$ for $k < \sqrt[4]{n}$ for the k-Distinctness Problem.

3 Euler Characteristic and Betti Numbers

The number of connected components is also known as the 0th Betti number. Defining higher Betti numbers is beyond the scope of this lecture, but we will mention that the method of connected components can be generalized to higher Betti numbers and Euler characteristics (another useful notion from topology). As with the number of connected components, log of Euler characteristic is a lower bound on the computation time. Higher Betti numbers are used similarly for obtaining lower bounds. The generalized method gives super-linear lower bounds for a broader class of problems.