Lecture 18:

Lower bound techniques
How to prove lower bounds?

easy? sublinear time algorithms see very little of input.
difficult? sublinear time algorithms are usually randomized
How to prove lower bounds?

Easy? Sublinear time algorithms see very little of input.

Difficult? Sublinear time algorithms are usually randomized.

Useful lower bound tool:

Yao's Principle: Given distribution $D$ on union of "positive" (Yes, Pass) instances & "negative" (No, Fail) inputs, such that any deterministic algorithm of query complexity $\leq t$ is incorrect with prob $\geq 1/3$ on inputs chosen from $D$, then $t$ is a lower bound on randomized query complexity.

(Proof Omitted)
Game theoretic view:

Alice selects deterministic alg $A$ and Bob selects input $x$.

A selects randomized algorithm $\iff A$ picks a random deterministic algorithm (fixed once you set random bits).

Von Neuman's minimax $\Rightarrow$ when $A$ randomized, a randomized Bob can do just as well as when $A$ deterministic.

$\Rightarrow$ if want to show lower bound, need only show distribution on inputs that is bad for every deterministic algorithm.
Example application of Yao's method:

\[ \mathbb{PAL} = \{ w \mid w \text{ is } w = VV^R u u^R \} \]

Concatenation of palindromes

E.g. $00011111110001110001110$

\[ \frac{w}{w} \quad \frac{w}{w} \quad \frac{w}{w} \quad \frac{w}{w} \]

\[ V_1 v_2 \ldots V_k v_k v_k^{-1} \ldots v_2^{-1} u_1 u_2 \ldots u_{2k-1} \]

Note that testing $\mathbb{PAL} = \{ w \mid w = VV^R \}$ is trivial:

pick random $i$, if $w_i \neq w_{n-i}$ fail.
Thm. any property tester for 2PAL needs $\sqrt{n}$ queries.

e.g. if $d$ satisfies $\forall x \in 2PAL$, $\Pr[A(x) = \text{PASS}] \geq 2/3$

+ $\forall x \in \text{far from 2PAL}$, $\Pr[A(x) = \text{FAIL}] \geq 2/3$

then $A$ makes $\Omega(\sqrt{n})$ queries.

Pf. Plan: give distribution on inputs that is hard for all algorithms using $o(\sqrt{n})$ queries.

Yao $\Rightarrow$ randomized l.b. of $\Omega(\sqrt{n})$
Distribution on "Fail" inputs:

\[ F = \text{random string of distance } \geq \varepsilon n \text{ from } 2PAL \]

Distribution on "Pass" inputs: \( (wlog \text{ assume } 6/n) \)

\[ P = \begin{cases} 
1. \text{ pick } k \in R \left[ \frac{n}{6} + 1, \frac{n}{3} \right] \\
2. \text{ pick random } v, u \text{ s.t. } |v| = k, |u| = \frac{n-k}{2} \\
3. \text{ output } v^ru^u \end{cases} \]

Note: Some strings can be generated via multiple K's. E.g. 0000...0

Bad Distribution:

\[ D = \begin{cases} 
\text{flip coin} \\
\text{if } H \text{ output according to } F \\
\text{else} \end{cases} \]
Assume deterministic algorithm $A$ s.t.

\[ \forall x \in 2\mathrm{PAL}, \quad \Pr[A(x) = \text{PASS}] \geq \frac{2}{3} \]

\[ \forall x \text{ far from } 2\mathrm{PAL}, \quad \Pr[A(x) = \text{FAIL}] \geq \frac{2}{3} \]

We can make $o\left(\frac{1}{\delta^2}\right)$ queries. 

* Wlog all leaves have depth $t$. 

- $\leq 2^t$ root-leaf paths

We can calculate prob of reaching each leaf given input dist.

Describe via query tree:

(for inputs of size $n$)

Query location

next query

if see 0

if see 1

\[
\begin{array}{c}
  i_0 \quad i_1 \\
  i_{10} \quad i_{11} \\
  i_{100} \quad i_{101} \\
  \vdots \ & \vdots \\
  \end{array}
\]

Query tree:
For each leaf \( l \):

\[
E^-(l) = \{ w \in \mathcal{Q} \} \text{ s.t. } \text{dist}(w, 2\text{PAL}) \geq 3n \quad \text{if } w \text{ reaches leaf } l^3
\]

\[
E^+(l) = \{ w \in \mathcal{Q} \} \text{ and } W \text{ reaches leaf } l^3
\]

Total error of \( \Delta \) on \( 0 \):

\[
= \sum_{l \text{ passing}} \Pr_{w \in E^-(l)}[w] + \sum_{l \text{ failing}} \Pr_{w \in E^+(l)}[w]
\]

\( \uparrow \text{ should Fail} \)

\( \uparrow \text{ should Pass} \)
For each leaf $l$:

For each leaf $l$:

\[ E^-(l) = \sum_{w \in \mathcal{I}^t} \text{st. } \text{dist}(w, \text{2PAL}) \geq En + w \text{ reaches leaf } L^l \]

\[ E^+(l) = \sum_{w \in \mathcal{I}^t} \text{st. } \text{dist}(w, \text{2PAL}) < En + w \text{ reaches leaf } L^l \]

Claim 1: if $t = \Theta(n)$, for all leaves $l$ at depth $t$

\[ \Pr_{D^t}[w \in E^-(l)] \geq \left(\frac{1}{2} - o(1)\right) 2^{-t} \]

Claim 2: if $t = \Theta(n)$, for all leaves $l$ at depth $t$

\[ \Pr_{D^t}[w \in E^+(l)] \geq \left(\frac{1}{2} - o(1)\right) 2^{-t} \]

Total error of $A$ on $D$:

\[ \sum_l \Pr_{\text{passing}} \left(\frac{1}{2} - o(1)\right) 2^{-t} + \sum_l \Pr_{\text{failing}} \left(\frac{1}{2} - o(1)\right) 2^{-t} \geq \frac{1}{2} - 6o(1) > > \frac{1}{3} \]
Claim 1: if \( t = o(n), \forall \ell \) at depth \( t \)

\[
P_r \left[ w \in E^{-\ell}(U) \right] \geq \left( \frac{1}{2} - o(1) \right) 2^{-t}
\]

Proof:

Plan:

- \( F \) is close to \( U \)

- \( U \) is uniformly distributed at each leaf
  (each \( \ell \) can have random bit, so go left/right with equal probability)

\[
\Rightarrow P_{w \in U} \left[ w \in E^{-\ell}(U) \right] = 2^{n-t} \leq 2^t
\]

But how much can distribution on leaves change using \( F \)?

(input size \( n \))

\[
|2 PAL_n| \leq 2^{n-1} \cdot \frac{n}{2} \quad \text{choice of } \lambda
\]

# words at distance \( \ell \) from \( 2 PAL_n \) \( \leq 2^{n-1} \cdot \frac{n}{2} \cdot \sum_{i=0}^{\ell} \binom{n}{i} \leq 2^{n/2 + \log(2) \cdot n} \)

Very few!!
so \( E^{-}(l) \approx 2^{n-t} - 2^{n/2 + 2\epsilon \log \frac{1}{\epsilon \nu}} = (1 - o(1)) 2^{n-t} \)

\[ \text{so strings in } U \text{ reaching } l \]
\[ \text{# words not in } F \]
\[ \text{assume } \epsilon \ll \frac{1}{8} \text{ so 1st term is } o(n) \text{ swamps this} \]

so \( \Pr_{b}[w \in E^{-}(l)] = \frac{1}{2} \Pr_{F}[w \in E^{-}(l)] \)

\[ = \frac{1}{2} \frac{|E^{-}(l)|}{2^n} \geq (\frac{1}{2} - o(1)) 2^{-t} \]
Claim 2. If \( t = O(\sqrt{n}) \), \( \forall \lambda \) at depth \( t \):
\[
\Pr_\theta [ w \in E^+(\lambda)] \geq \left( \frac{1}{2} - o(1) \right) 2^{-t}
\]

Proof. Plan: for every fixed set of \( o(\sqrt{n}) \) queries, lots of strings in 2PAL follow the path.

- How many strings agree with leaf \( \lambda \)? \( 2^{n-t} \)
- How many \( n \)-bit strings in 2PAL agree with leaf \( \lambda \)?
  \[
  \geq 2^{\frac{n}{2} - t} \quad ???
  \]

Difficulty:

Fix \( k = 10 \): should see same value at:
- \( 1/10 \)
- \( 2/9 \)
- \( 3/8 \)

Lots of "dependencies"}

- \( F = \) random string of distance \( \geq 3n \) from 2PAL
- \( P = \{ 1. \ \text{pick} \ K \in [\frac{n}{2}, \frac{n}{3}] \\
\quad 2. \ \text{pick} \ \text{random} \ \nu, \mu \quad \text{s.t.} \quad |\nu| = k \quad |\mu| = \frac{a-k}{2} \\
\quad 3. \ \text{output} \ \nu \mu \wedge u \}
\)

- \( D = 0 \) if \( H \) output according to \( F \)
  - else "" "" "" "" "" "" "" "" "" ""

Maybe no string follows path?
but \( k \) is picked randomly! \( k \in \left[ \frac{n}{6} + 1, \ldots, \frac{n}{3} \right] \)

hope: paths that pair up dependent queries for one \( k \) will do badly on most others?

Consider leaf \( l \),

\[ Q_l \leftarrow \text{indices queried along way} \]

pair \( q_1, q_2 \in Q_l \), at most 2 choices of \( k \) "pair" them:

\[ \Rightarrow \text{choices of } k \text{ s.t. } \frac{n}{6} - 2 \left( \frac{1}{2} \right) = (1 - \delta(1)) \left( \frac{n}{6} \right) \]

"Good" \( k \)
Claim 2: If $\epsilon = O(\sqrt{n})$, $\forall \ell$ at depth $t$

$$\Pr_p[\ell \in E^+(\ell)] \geq (\frac{1}{2} - o(1)) 2^{-t}$$

Proof:

Plan: For every fixed set of $o(\sqrt{n})$ queries, lots of strings in 2PAL follow the path.

- How many strings agree with leaf $\ell$? $2^{n-t}$
- How many $n$-bit strings in 2PAL agree with leaf $\ell$?
  $$\geq \frac{n}{6} - 2^{\frac{n}{2}} = (1 - o(1)) \left(\frac{n}{6}\right)$$
  "Good" $k$

So

$$\Pr_p[\ell \in E^+(\ell)] \geq \sum_{w} \Pr_p[w|k] \cdot \Pr[k] \cdot \Pr[\text{choose } k] - 1_{\ell \in E^+(\ell)}$$

$$\geq \left(\frac{1}{6}\right)^{n/2} \cdot \left(1 - o(1)\right) \frac{n}{6} - t \geq \left(1 - o(1)\right) 2^{-t}$$

$F$ = random string of distance $\geq \epsilon n$ from 2PAL

$$P = \left\{\begin{array}{l}
1. \text{ pick } k \in \mathbb{R} \left[\frac{n}{6}, \frac{n}{3}\right] \\
2. \text{ pick random } v, u \text{ st. } |v - k| |u - k| = \frac{n - k}{2} \\
3. \text{ output } vv \oplus uu
\end{array}\right.$$