Adaptivity does not help: Nearly tight lower bounds for Boolean monotonicity testing

Xi Chen, Mark Chen, Hao Cui,
William Pires, Jonah Stockwell

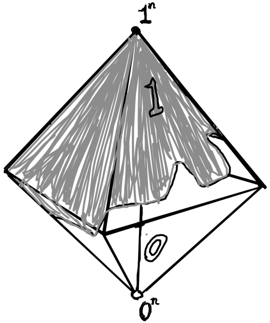
Columbia University

Adaptivity does not help: Nearly tight lower bounds for Boolean monotonicity testing
… or *How to hide anti-dictators*

Xi Chen, Mark Chen, Hao Cui,
William Pires, Jonah Stockwell

Columbia University

Given a Boolean function $f : \{0,1\}^n \to \{0,1\}$:



monotone if $f(x) \leq f(y)$ for all $x, y \in \{0,1\}^n$ with $x \preceq y$.

Given black-box access to an $f : \{0,1\}^n \to \{0,1\}$ that is not monotone, how many queries does it take to find a violating pair:

$$(x,y) \text{ with } x \preceq y \text{ but } f(x) = 1 \text{ and } f(y) = 0$$

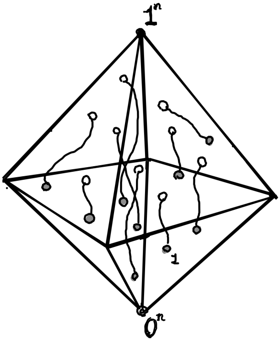
Given black-box access to an $f : \{0,1\}^n \to \{0,1\}$ that is far from monotone, how many queries does it take to find a violating pair:

$$(x, y) \text{ with } x \preceq y \text{ but } f(x) = 1 \text{ and } f(y) = 0$$

far: at least 1% *of entries flipped to become monotone*

one-sided monotonicity testing

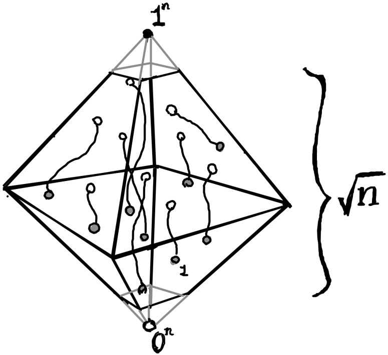
Far-from-monotone Functions



Equivalent Characterization [Fischer et al. 02]

f is far from monotone iff there exist $\Omega(2^n)$ disjoint violating pairs.

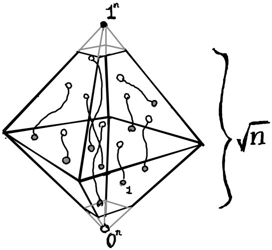
Middle layers



Equivalent Characterization [Fischer et al. 02]

f is far from monotone iff there exist $\Omega(2^n)$ disjoint violating pairs.

Given f : middle layers $\to \{0,1\}$ with $\Omega(2^n)$ disjoint violating pairs:



How many queries are needed to find a violating pair?

Property testing

Monotonicity testing:

- [Goldreich, Goldwasser, Lehman, Ron and Samordinsky 98]
- [DGL+99, FLN+02, HK08, BCGM12, RRS+12, BBM12, BRY13, CS13, CS14, CST14, KMS15, CDST15, BB16, CWX 17, CS 19]
- More papers on real-valued functions over the hypergrid [HK07, AC06, SS08, FR10, BGJ+12, CS13, BRY14a, BRY14b, BCS18, BCS20, BKR24, HY20, BKKM22, BCS23, BCS25].

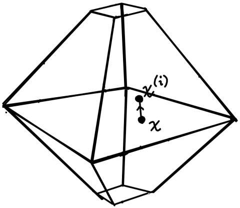
This Talk

Theorem (CCCPS 25)

Any adaptive algorithm requires $n^{0.5-c}$ queries, for any constant $c > 0$.

Nearly closes the gap of $O(\sqrt{n})$ vs $\Omega(n^{1/3})$
Works against adaptive, two-sided algorithms.

Edge tester



Pick a random edge $(x, x^{(i)})$ and query both points.

Theorem (Goldreich et al. 98)

If f is far from monotone, then

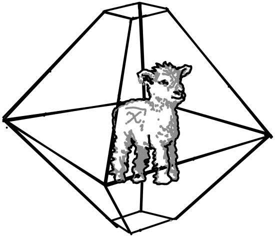
$$\mathbf{E}_{\mathbf{x}\sim\{0,1\}^n}\big[\textit{num of violating edges adj. to } \mathbf{x}\big] = \Omega(1).$$

So $\Omega(2^n)$ violating edges and $O(n)$ queries suffice.

Directed analogue of a folklore isoperimetric inequality:

$$\mathbf{E}_{\mathbf{x}\sim\{0,1\}^n}\big[\text{num of bichromatic edges adj. to } \mathbf{x}\big] = \Omega\big(\text{var}(f)\big).$$

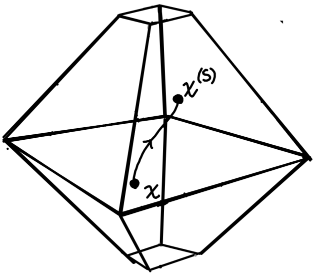
Anti-dictatorship functions



$\Omega(n)$ queries are required

~~Left to the audience as an exercise~~

Pair tester



Let ℓ be a parameter. Draw a random point x, flip a set S of ℓ random 0's of x to 1, and query both points x and $x^{(S)}$.

Theorem (Khot, Minzer and Safra 15)

If f is far from monotone, then

$$\mathbf{E}_{\mathbf{x}\sim\{0,1\}^n}\left[\sqrt{\textit{num of violating edges adj. to } \mathbf{x}}\right] = \Omega(1),$$

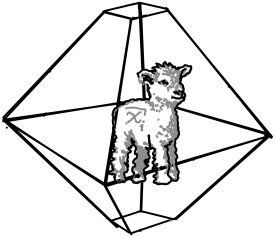
which led to an $O(\sqrt{n})$ upper bound for the path tester.

Directed analogue of an isoperimetric inequality of [Talagrand 93]:

$$\mathbf{E}_{\mathbf{x}\sim\{0,1\}^n}\left[\sqrt{\text{num of violating edges adj. to } \mathbf{x}}\right] = \Omega\big(\mathrm{var}(f)\big).$$

- [Chakrabarty and Seshadhri 13]: $O(n^{7/8})$
- [Chen, Servedio and Tan 14]: $O(n^{5/6})$

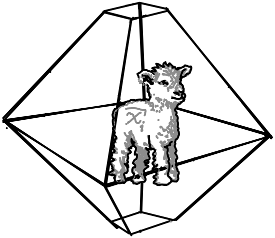
Nonadaptive lower bounds



Theorem (Fischer, Lehman, Newman, Raskhodnikova, Rubinfeld and Samorodnitsky 02)

Any nonadaptive algorithm that finds a violating pair needs $\sqrt{n}$ queries.

But not against adaptive enemies ...



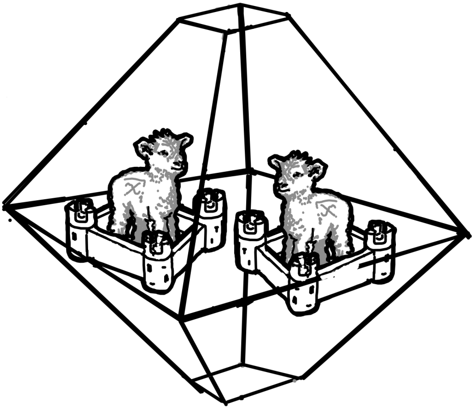
Binary search or variance estimation

Adaptive Lower Bounds

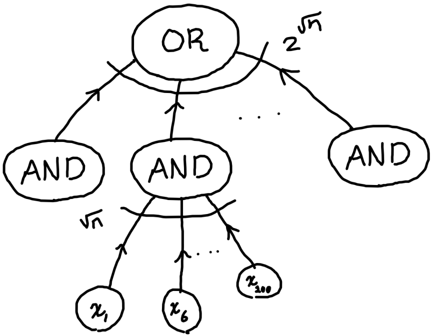
Adaptive lower bounds, or ... how to hide anti-dictators:

1. Talagrand Functions [Belovs and Blais 16]: $\Omega(n^{1/4})$
2. Two-level Talagrand Functions [C, Waingarten and Xie 17]: $\Omega(n^{1/3})$
3. This work: *Multi-level Talagrand Functions*

Blueprint of [Belovs and Blais 16]



Talagrand's function



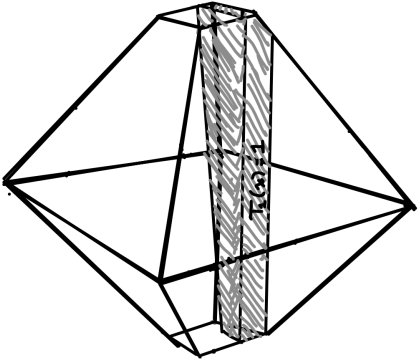
Let $s = \sqrt{n}$. OR of 2^s terms: $T_1, \ldots, T_{2^s}$

Each term T_i: s variables, sampled independently and uniformly.

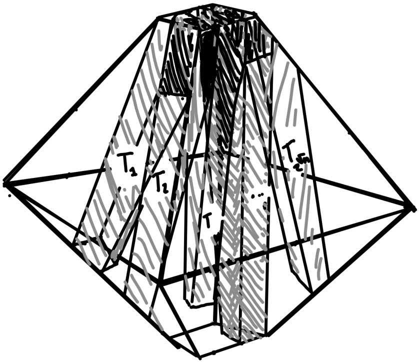
Why setting $s = \sqrt{n}$?

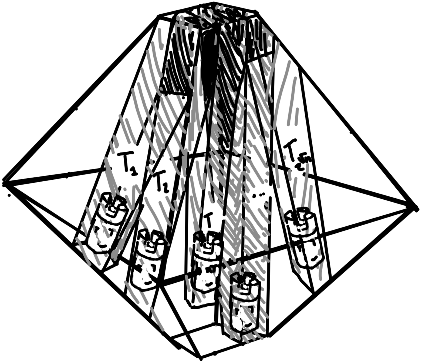
Largest possible to make the function interesting: Fixing a $z \in \{0,1\}^n$

$$\Pr_{T}\left[T(z)=1\right] = \left(\frac{|z|}{n}\right)^{\sqrt{n}} = \left(\frac{(n/2) \pm \sqrt{n}}{n}\right)^{\sqrt{n}} = \left(1 \pm \frac{1}{\sqrt{n}}\right)^{\sqrt{n}} \cdot \frac{1}{2^{\sqrt{n}}} = \Theta(1) \cdot \frac{1}{2^{\sqrt{n}}}.$$



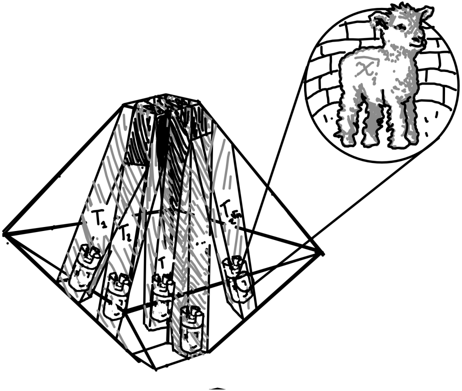
T1(x)=1


T_1
T_2
T
$T_{2^{\sqrt{n}}}$



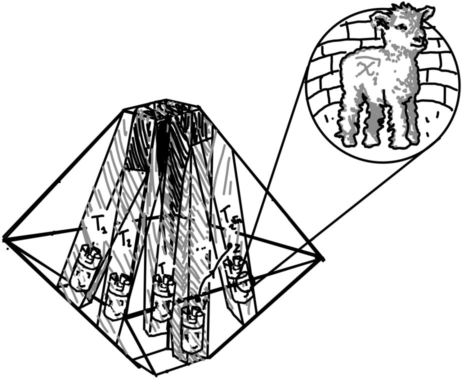
The castles!

Blueprint of [Belovs and Blais 16]



1. The function is far from monotone; and
2. The only violations are (x, y) that (1) lie in the same castle and (2) do not agree on the anti-dictator of this castle.

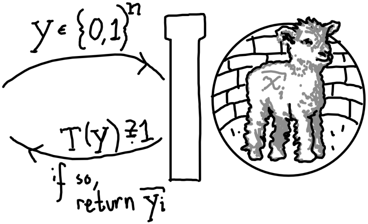
Blueprint of [Belovs and Blais 16]



1. The function is far from monotone; and
2. The only violations are (x, y) that (1) lie in the same castle and (2) do not agree on the anti-dictator of this castle.

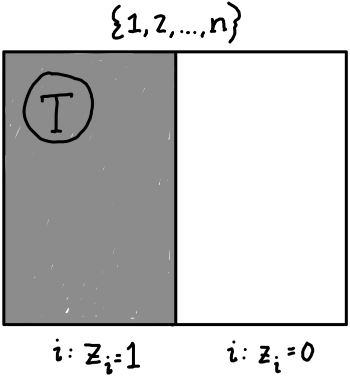
The castle game

Given a point z with $T(z) = 1$ for free:

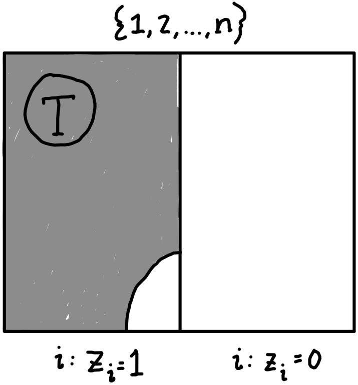


Goal: Find y, y' with $T(y) = T(y') = 1$ and $y_i \neq y'_i$.

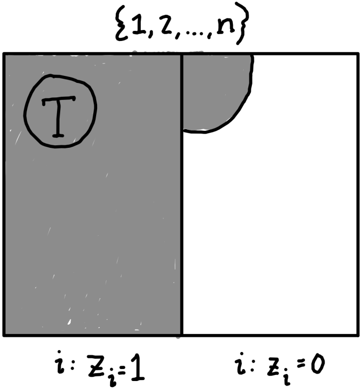
$\Omega(\sqrt{n})$?



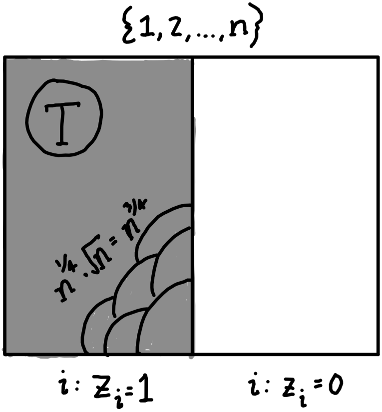
$\Omega(\sqrt{n})$?



$\Omega(\sqrt{n})$?

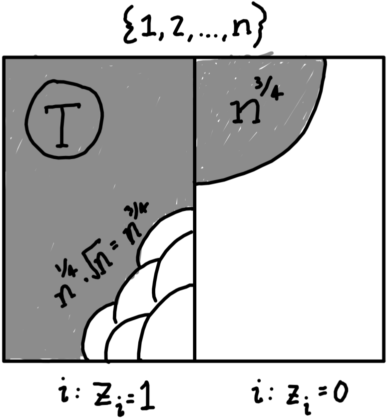


Quadratic speedup



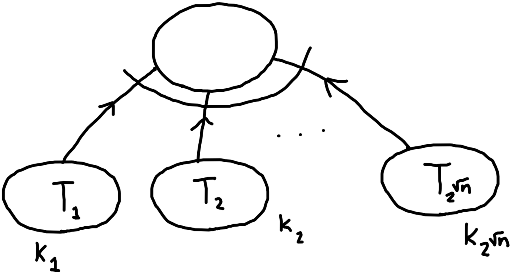
Stage 1: Use $n^{1/4}$ queries to find $\Omega(n^{3/4})$ variables not in T.

Quadratic speedup

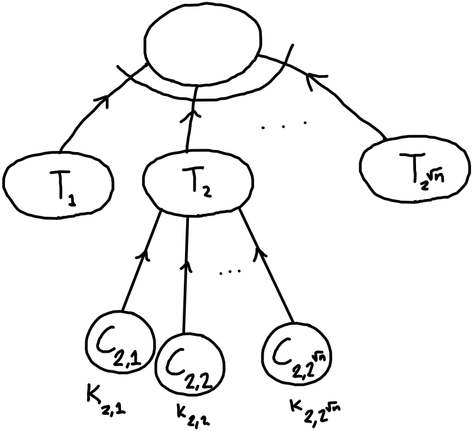


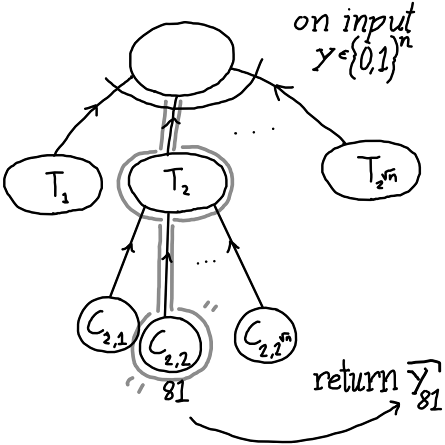
Stage 2: Use $n^{1/4}$ queries to search through the white region.

Construction of [Chen, Waingarten and Xie 17]



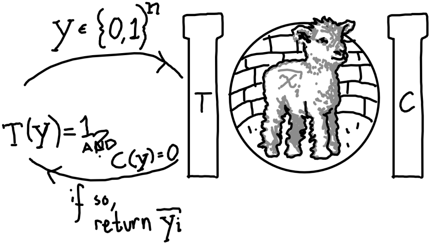
Construction of [Chen, Waingarten and Xie 17]




on input
$y \in \{0,1\}^n$
T_1
T_2
$T_{2^{\sqrt{n}}}$
$C_{2,1}$
$C_{2,2}$
$C_{2,2^{\sqrt{n}}}$
81
return $\overline{y_{81}}$

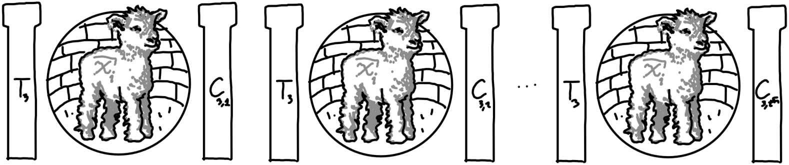
The new castle game

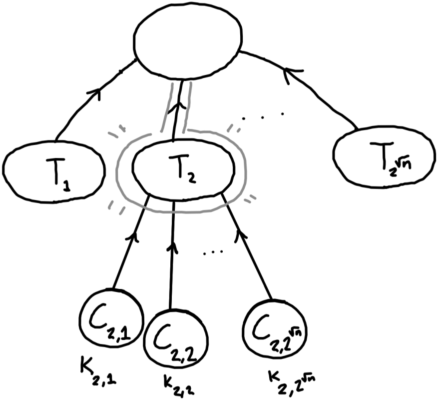
Given a point z with $T(z) = 1$ and $C(y) = 0$ for free:

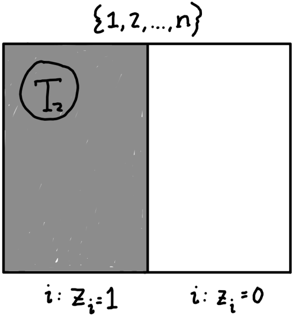


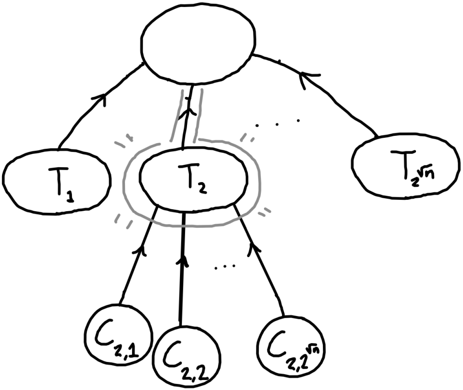
Goal: Find y, y' with $T(y) = T(y') = 1$, $C(y) = C(y') = 0$ and $y_i \neq y'_i$.

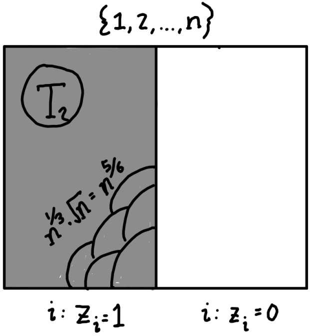
Quadratic speedup strikes back ...

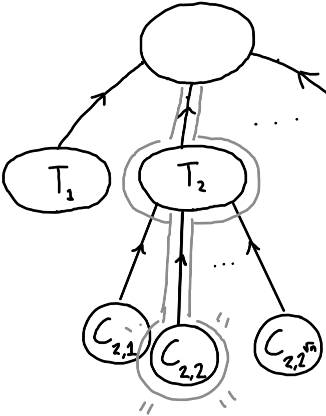



T_1
T_2
$T_{2^{\sqrt{n}}}$
$C_{2,1}$
$C_{2,2}$
$C_{2,2^{\sqrt{n}}}$
$K_{2,1}$
$K_{2,2}$
$K_{2,2^{\sqrt{n}}}$

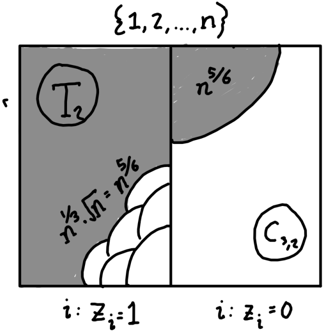

$\{1, 2, \dots, n\}$
T_2
$i: z_i = 1$
$i: z_i = 0$


T1
T2
T2√n
C2,1
C2,2
C2,2√n

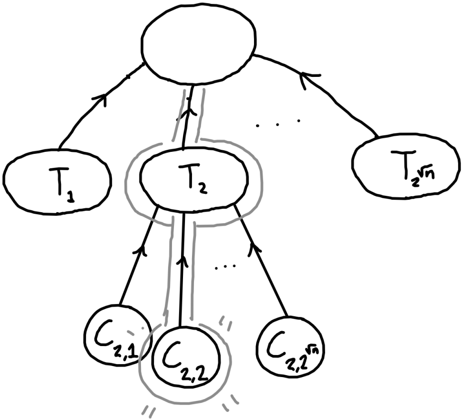

{1, 2, …, n}
T2
n1/3 · √n = n5/6
i: zi = 1
i: zi = 0

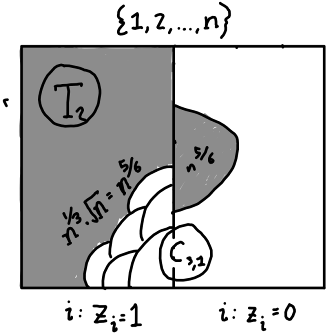


T_1
T_2
$T_{2\sqrt{n}}$
$C_{2,1}$
$C_{2,2}$
$C_{2,2\sqrt{n}}$



$\{1, 2, \ldots, n\}$
T_2
$n^{5/6}$
$n^{1/3} \cdot \sqrt{n} = n^{5/6}$
$C_{3,2}$
$i : z_i = 1$
$i : z_i = 0$


T
T
T
T
C
C
C
...
...


{1, 2, ..., n}
T
C
i : Z_i = 1
i : Z_i = 0

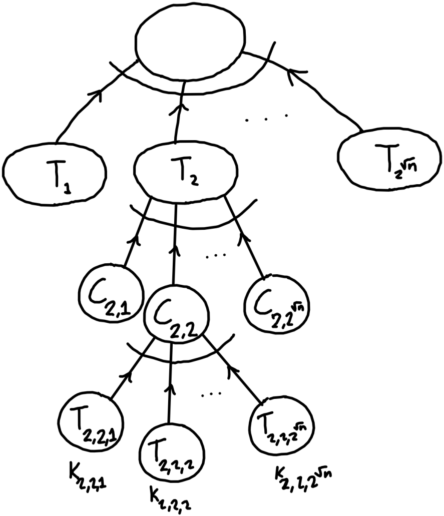
Summary

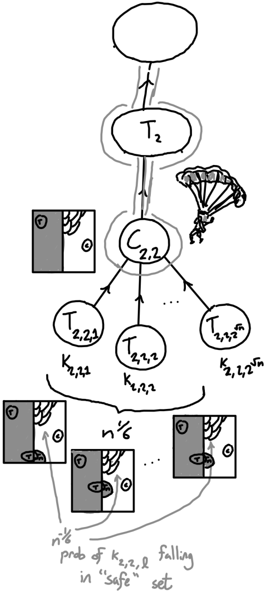
- one-level: $q^2\sqrt{n} \Rightarrow q = \Omega(n^{1/4})$
- two-level: $q^{3/2}\sqrt{n} \Rightarrow q = \Omega(n^{1/3})$
  - q queries to obtain $q\sqrt{n}$ safe variables for the term
  - for each clause below, $\sqrt{q}$ queries suffice to search through $q\sqrt{n}$
  - repeat for $\sqrt{q}$ clauses below to cover $q^{3/2}\sqrt{n}$ in total

More levels?

- one-level: $q^2\sqrt{n} \Rightarrow q = \Omega(n^{1/4})$
- two-level: $q^{3/2}\sqrt{n} \Rightarrow q = \Omega(n^{1/3})$
  - q queries to obtain $q\sqrt{n}$ safe variables for the term
  - for each clause below, $\sqrt{q}$ queries suffice to search through $q\sqrt{n}$
  - repeat for $\sqrt{q}$ clauses below to cover $q^{3/2}\sqrt{n}$ in total
- three-level: $q^{4/3}\sqrt{n} \Rightarrow q = \Omega(n^{3/8})$?

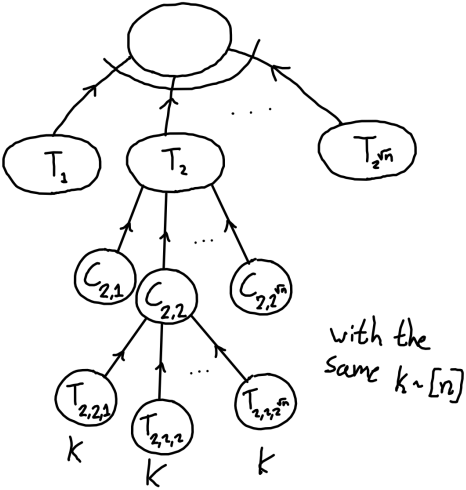
  $\ldots$
- ℓ-level: $q^{(\ell+1)/\ell}\sqrt{n} \Rightarrow q = \Omega(n^{\ell/2(\ell+1)})$?


T1
T2
T2√n
C2,1
C2,2
C2,2√n
T2,2,1
T2,2,2
T2,2,2√n
K2,2,1
K2,2,2
K2,2,2√n



T_2
$C_{2,2}$
$T_{2,2,1}$
$T_{2,2,2}$
$T_{2,2,2^{\sqrt{n}}}$
$K_{2,2,1}$
$K_{2,2,2}$
$K_{2,2,2^{\sqrt{n}}}$
$n^{1/6}$
$n^{-1/6}$
prob of $K_{2,2,\ell}$ falling in "safe" set

The fix after eight years ...



More levels!

- one-level: $q^2\sqrt{n} \Rightarrow q = \Omega(n^{1/4})$
- two-level: $q^{3/2}\sqrt{n} \Rightarrow q = \Omega(n^{1/3})$
  - q queries to obtain $q\sqrt{n}$ safe variables for the term
  - for each clause below, $\sqrt{q}$ queries suffice to search through $q\sqrt{n}$
  - repeat for $\sqrt{q}$ clauses below to cover $q^{3/2}\sqrt{n}$ in total
- three-level: $q^{4/3}\sqrt{n} \Rightarrow q = \Omega(n^{3/8})$

  $\ldots$
- ℓ-level: $q^{(\ell+1)/\ell}\sqrt{n} \Rightarrow q = \Omega(n^{\ell/2(\ell+1)})$

Open Problems

- Our lower bounds: $\Omega(n^{0.5-c})$ for any $c > 0$
- [Minzer 26]: $\Omega(n^{0.5-o(1)})$ by a distance amplification lemma
- Close the gap with $\tilde{\Omega}(\sqrt{n})$?
- Reduce the exponential gap between upper $\left(2^{\tilde{O}(\sqrt{n})}\right)$ and lower bounds $\left(2^{\Omega\left(n^{1/4}\right)}\right)$ for tolerant testing of monotonicity and many other related problems.
  In particular, can estimating distance from monotonicity be done faster than learning?

Thank you!

Xi Chen
Hao Cui
William Pires
Jonah Stockwell*

Columbia University

*To whom the art credit is due; an undergrad applying for a PhD position in the coming cycle!!!