Strongly Exponential Lower Bounds for Monotone Computation

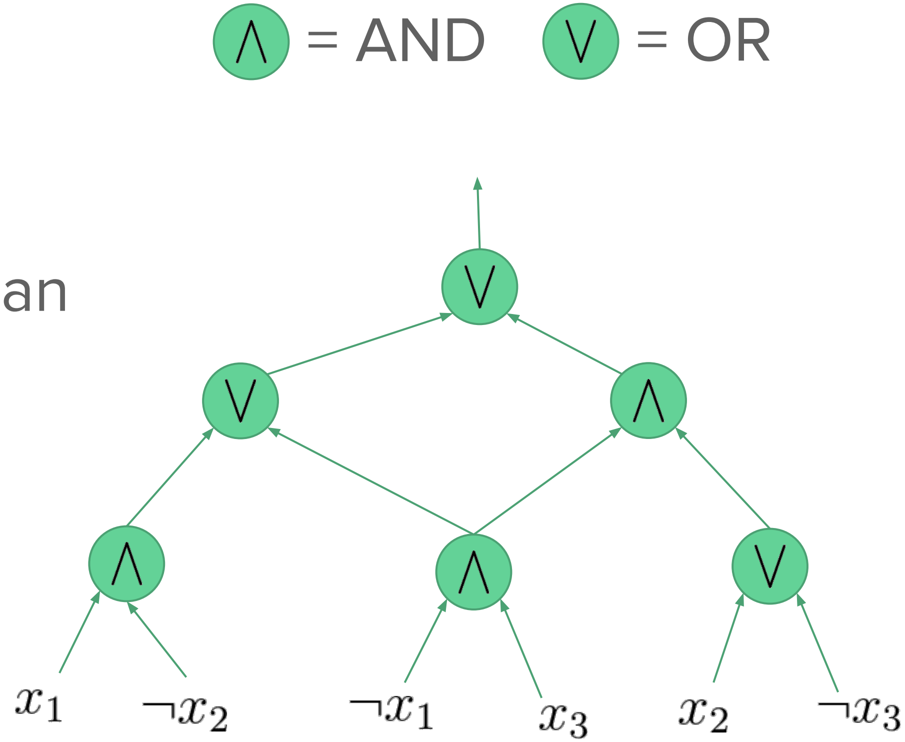
Toniann Pitassi and **Robert Robere**
Department of Computer Science
University of Toronto

STOC 2017
Montréal, Canada

Boolean Circuits

Basic model for computing boolean functions $f : \{0,1\}^n \to \{0,1\}$

Assume fan-in 2, and a basis of AND, OR, NOT gates.



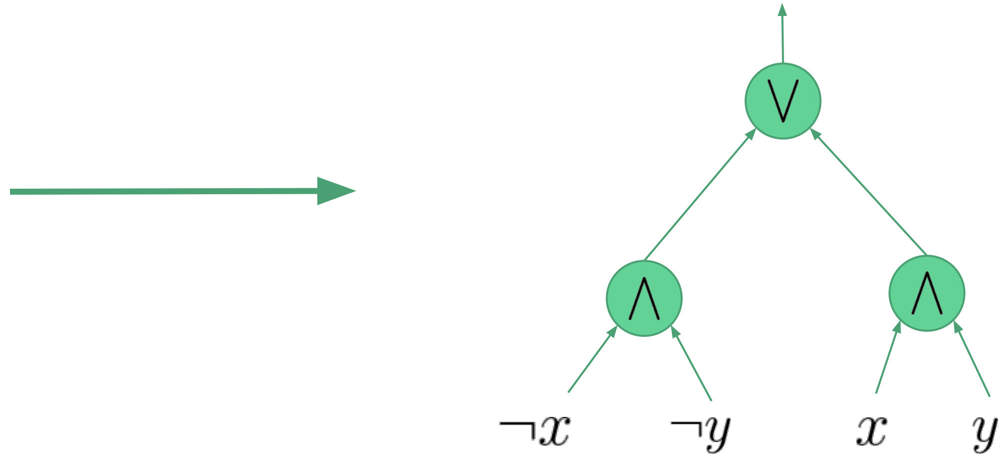
Central Question.

What boolean functions are hard to compute?

Boolean Circuits

Every $f:\{0,1\}^n \to \{0,1\}$ has a circuit of size $O(n2^n)$.

x	y	$f(x,y)$
0	0	1
0	1	0
1	0	0
1	1	1



Theorem. [Lupanov 58] Every boolean function on n bits can be computed by a circuit with $(1+o(1))\frac{2^n}{n}$ gates (!)

Boolean Circuits

Theorem. [Lupanov 58] Every boolean function on n bits can be computed by a circuit with $(1 + o(1))\frac{2^n}{n}$ gates (!)

Theorem. [Shannon 1949] For every n, all but an exponentially small fraction of boolean functions on n bits require circuits with $\Omega\left(\frac{2^n}{n}\right)$ gates.

Proof. Simple counting argument (non-constructive).

Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

NO!

Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

NO!

Complexity Measure	**Circuit Type**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
P	Circuits		$2^n/n$ [S. 49]

Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

NO!

Complexity Measure	**Circuit Type**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
P	Circuits	$5n - o(n)$ [IM. 02]	$2^n/n$ [S. 49]

Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

NO!

Complexity Measure	**Circuit Type**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
P	Circuits	$5n - o(n)$ [IM. 02]	$2^n/n$ [S. 49]
NC^1	Formula		

Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

NO!

Complexity Measure	**Circuit Type**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
P	Circuits	$5n - o(n)$ [IM. 02]	$2^n / n$ [S. 49]
NC^1	Formula		$2^n / \log n$ [RS. 42]

Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

NO!

Complexity Measure	**Circuit Type**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
P	Circuits	$5n - o(n)$ [IM. 02]	$2^n/n$ [S. 49]
NC^1	Formula	$n^{3-o(1)}$ [H. 98]	$2^n/\log n$ [RS. 42]

Boolean Circuits (Lower Bounds)

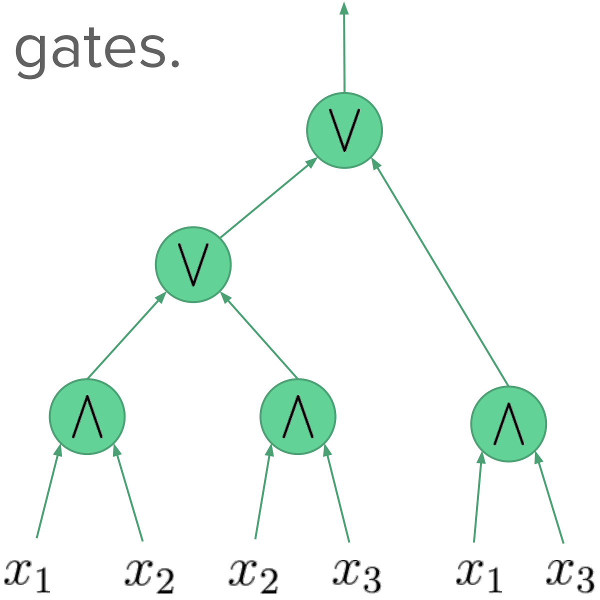
Do we have any **explicit** examples of hard boolean functions?

NO!

Complexity Measure	**Circuit Type**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
P	Circuits	$5n - o(n)$ [IM. 02]	$2^n/n$ [S. 49]
NC^1	Formula	$n^{3-o(1)}$ [H. 98]	$2^n/\log n$ [RS. 42]
L	Switching Networks	$n^2/\log n$ [N. 66]	$2^n/n$ [S. 49]
Mod_p L	Span Programs	$n \log n$ [KW. 91]	**GF(2)** $\sqrt{2^{n+1}}$ [N.62]
CC	Comparator Circuits	$n \log n$ [KLMPSS. 95]	$2^n/n$ [S. 49]

Monotone Circuit Complexity

A circuit is *monotone* if it does not use NOT gates.

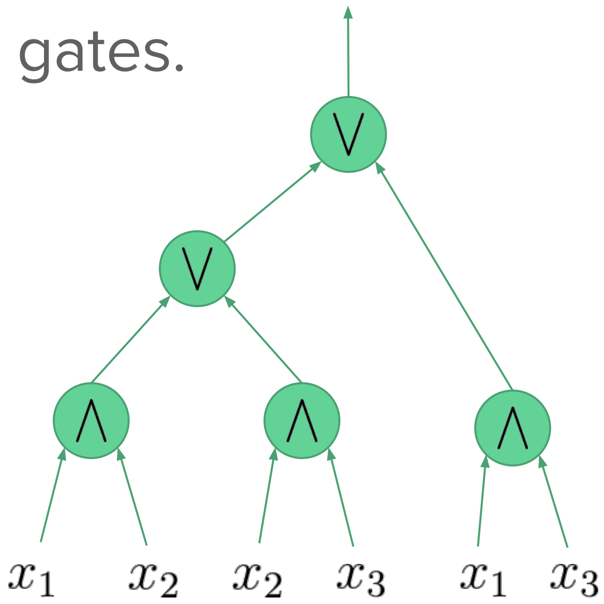


Monotone Circuit Complexity

A circuit is *monotone* if it does not use NOT gates.

A function $f : \{0,1\}^n \to \{0,1\}$ is monotone if

$$x \leq y \implies f(x) \leq f(y)$$



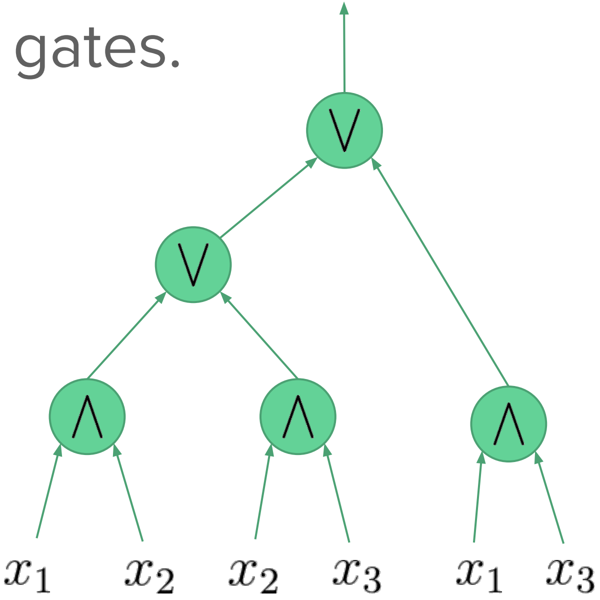
Monotone Circuit Complexity

A circuit is *monotone* if it does not use NOT gates.

A function $f : \{0,1\}^n \rightarrow \{0,1\}$ is monotone if

$$x \leq y \implies f(x) \leq f(y)$$

Monotone circuits have a number of applications in cryptography, proof complexity, communication theory ....



Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

YES!

Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

YES!

Complexity Measure	Circuit Type (MONOTONE)	Strongest Lower Bound (Explicit)	Strongest Lower Bounds (Non-Explicit)
mP	Circuits		$2^n/n^{3/2}$ [U, P 76]

Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

YES!

Complexity Measure	Circuit Type (MONOTONE)	Strongest Lower Bound (Explicit)	Strongest Lower Bounds (Non-Explicit)
mP	Circuits	$2^{\Omega((n/\log n)^{1/3})}$ [HR 01]	$2^n/n^{3/2}$ [U, P 76]

Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

YES!

Complexity Measure	Circuit Type (MONOTONE)	Strongest Lower Bound (Explicit)	Strongest Lower Bounds (Non-Explicit)
mP	Circuits	$2^{\Omega((n/\log n)^{1/3})}$ [HR 01]	$2^n/n^{3/2}$ [U, P 76]
mNC^1	Formula	$2^{\Omega(n/\log n)}$ [GP 14]	$2^n/\sqrt{n}\log n$

Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

YES!

Complexity Measure	**Circuit Type (MONOTONE)**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
mP	Circuits	$2^{\Omega((n/\log n)^{1/3})}$ [HR 01]	$2^n/n^{3/2}$ [U, P 76]
mNC^1	Formula	$2^{\Omega(n/\log n)}$ [GP 14]	$2^n/\sqrt{n}\log n$
mL	Switching Networks	$2^{\Omega(\sqrt{n/\log n})}$ [GP 14]	$2^n/n^{3/2}$ [U, P 76]

Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

YES!

Complexity Measure	**Circuit Type (MONOTONE)**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
mP	Circuits	$2^{\Omega((n/\log n)^{1/3})}$ [HR 01]	$2^n/n^{3/2}$ [U, P 76]
mNC^1	Formula	$2^{\Omega(n/\log n)}$ [GP 14]	$2^n/\sqrt{n}\log n$
mL	Switching Networks	$2^{\Omega(\sqrt{n/\log n})}$ [GP 14]	$2^n/n^{3/2}$ [U, P 76]
$\mathrm{mSPAN}_{\mathbb{R}}$	Real Span Programs	$2^{\Omega(n^{1/7})}$ [**RP**RC 16]	
mCC	Comparator Circuits	$2^{\Omega(n^{1/7})}$ [**RP**RC 16]	$2^n/n^{3/2}$ [U, P 76]

Monotone Boolean Circuits (Lower Bounds)

Do we have any **explicit** examples of hard boolean functions?

YES!

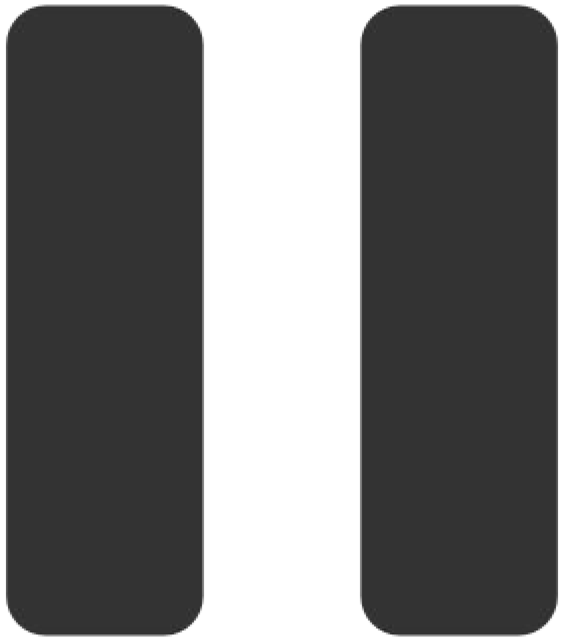
Complexity Measure	**Circuit Type (MONOTONE)**	**Strongest Lower Bound (Explicit)**	**Strongest Lower Bounds (Non-Explicit)**
mP	Circuits	$2^{\Omega((n/\log n)^{1/3})}$ [HR 01]	$2^n/n^{3/2}$ [U, P 76]
mNC^1	Formula	$2^{\alpha n}$ [**PR 17**]	$2^n/\sqrt{n}\log n$
mL	Switching Networks	$2^{\alpha n}$ [**PR 17**]	$2^n/n^{3/2}$ [U, P 76]
$\mathrm{mSPAN}_{\mathbb{R}}$	Real Span Programs	$2^{\alpha n}$ [**PR 17**]	
mCC	Comparator Circuits	$2^{\alpha n}$ [**PR 17**]	$2^n/n^{3/2}$ [U, P 76]

Result

Main Theorem. There is a monotone boolean function f computable in **NP** (CSP-SAT) such that every monotone

1. formula,
2. switching network,
3. real span program, or
4. comparator circuit

computing f requires size $2^{\alpha n}$ for some universal constant $\alpha > 0$.



The Proof (A Flavor)

Columns labelled with $y \in f^{-1}(0)$

Rows labelled with $x \in f^{-1}(1)$

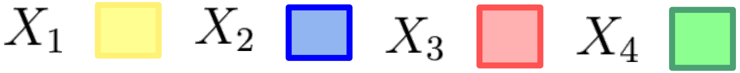
$f^{-1}(1) \times f^{-1}(0)$

Let $f : \{0,1\}^N \to \{0,1\}$ be a monotone boolean function.

$\text{KW-Search}^+(f) \subseteq f^{-1}(1) \times f^{-1}(0) \times [N]$

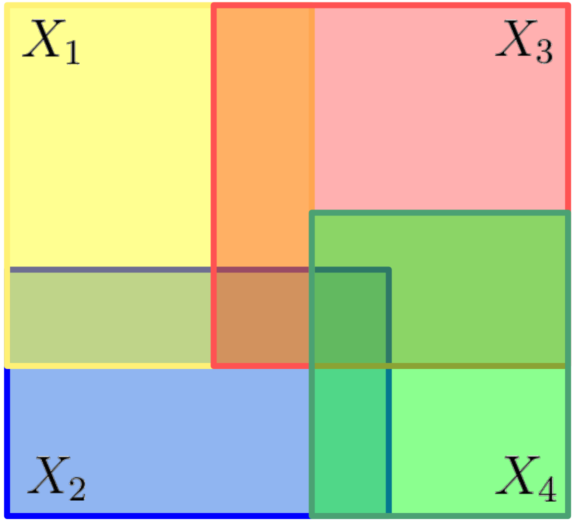
Input: $(x, y) \in f^{-1}(1) \times f^{-1}(0)$

Output: $i \in [N] \quad x_i = 1, y_i = 0$



Columns labelled with $y \in f^{-1}(0)$

Rows labelled with $x \in f^{-1}(1)$

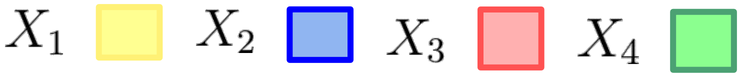


Let $f : \{0,1\}^N \to \{0,1\}$ be a monotone boolean function.

$\text{KW-Search}^+(f) \subseteq f^{-1}(1) \times f^{-1}(0) \times [N]$

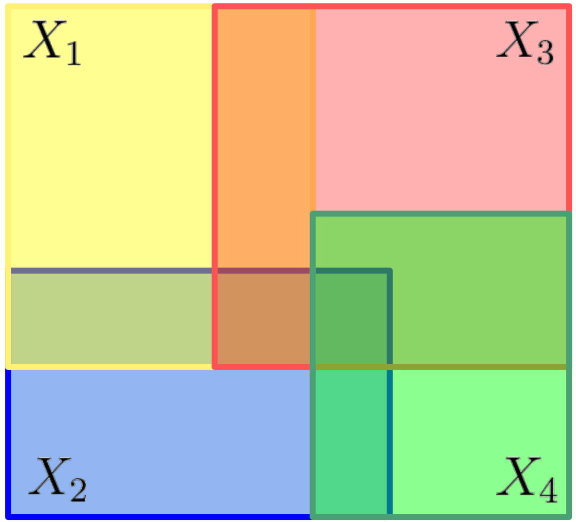
Input: $(x, y) \in f^{-1}(1) \times f^{-1}(0)$

Output: $i \in [N] \quad x_i = 1, y_i = 0$



Columns labelled with $y \in f^{-1}(0)$

Rows labelled with $x \in f^{-1}(1)$



Let $f : \{0,1\}^N \to \{0,1\}$ be a monotone boolean function.

$\text{KW-Search}^+(f) \subseteq f^{-1}(1) \times f^{-1}(0) \times [N]$

Input: $(x, y) \in f^{-1}(1) \times f^{-1}(0)$

Output: $i \in [N] \quad x_i = 1, y_i = 0$

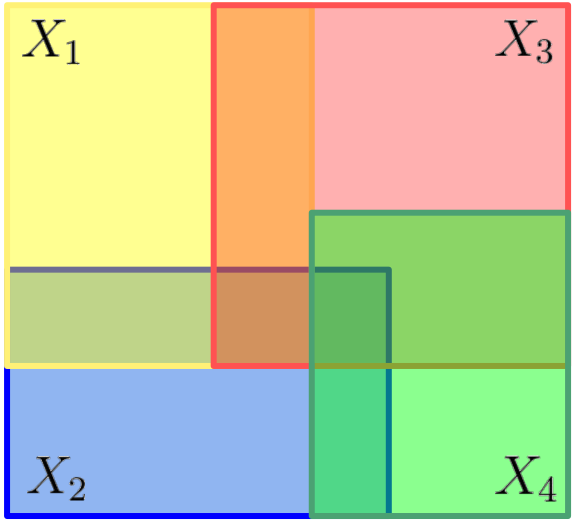
Theme: Complexity of $\text{KW-Search}(f) \approx$ Circuit Complexity of f

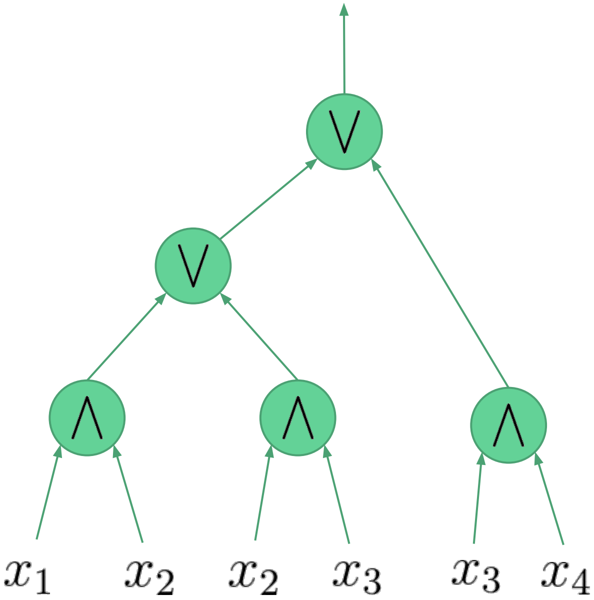
Example: Formulas

$\wedge$ = AND $\vee$ = OR

Columns labelled with $y \in f^{-1}(0)$

Rows labelled with $x \in f^{-1}(1)$





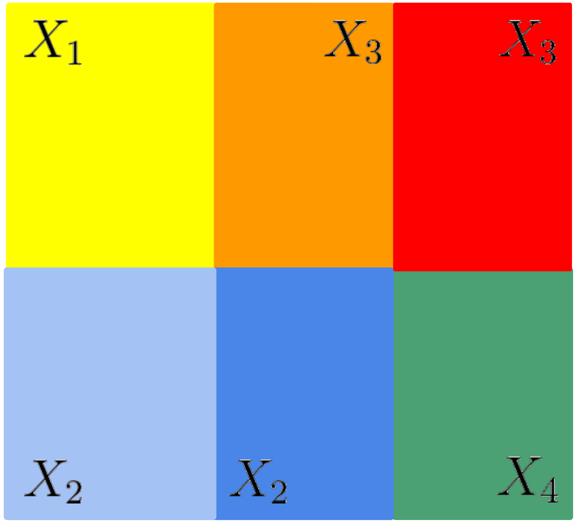
Theme: Complexity of $\text{KW-Search}(f) \approx$ Circuit Complexity of f

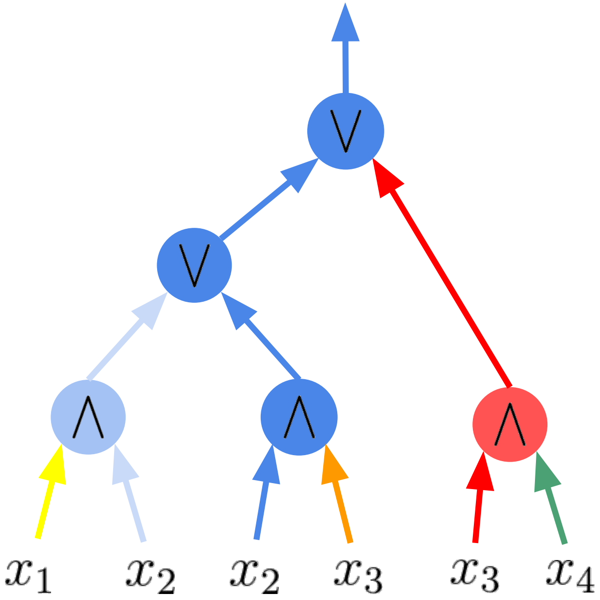
Example: Formulas

$\wedge$ = AND $\vee$ = OR

Columns labelled with $y \in f^{-1}(0)$

Rows labelled with $x \in f^{-1}(1)$



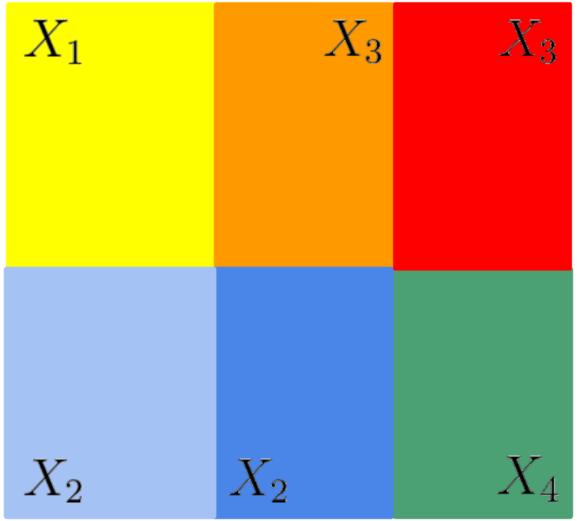


Lemma. [Khrapchenko 71] Formula for $f : \{0,1\}^n \to \{0,1\}$ with **s** leaves yields a partition of $f^{-1}(1) \times f^{-1}(0)$ into **s** mono. rectangles.

Let $\chi(f)$ denote the minimum number of rectangles in any monochromatic partition of $f^{-1}(1) \times f^{-1}(0)$

Columns labelled with $y \in f^{-1}(0)$

Rows labelled with $x \in f^{-1}(1)$



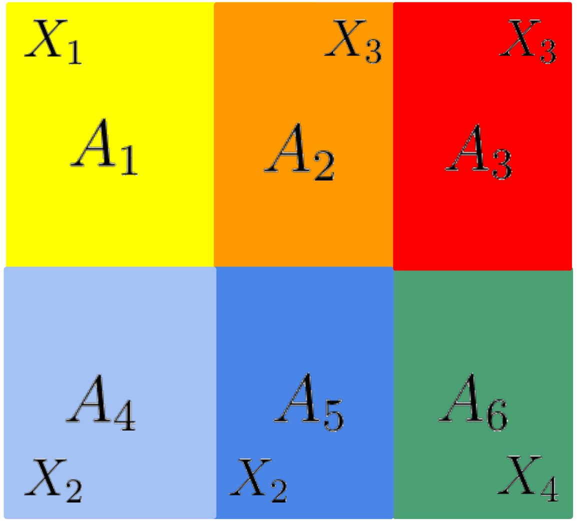
Idea [Razb. 90]: Use rank to lower bound $\chi(f)$!

Let $\chi(f)$ denote the minimum number of rectangles in any monochromatic partition of $f^{-1}(1) \times f^{-1}(0)$

Columns labelled with $y \in f^{-1}(0)$

A

Rows labelled with $x \in f^{-1}(1)$



Idea [Razb. 90]: Use rank to lower bound $\chi(f)$!

Let A be any $|f^{-1}(1)| \times |f^{-1}(0)|$ matrix over a field **F**.

$$A = \sum_{i=1}^{\chi(f)} A_i$$

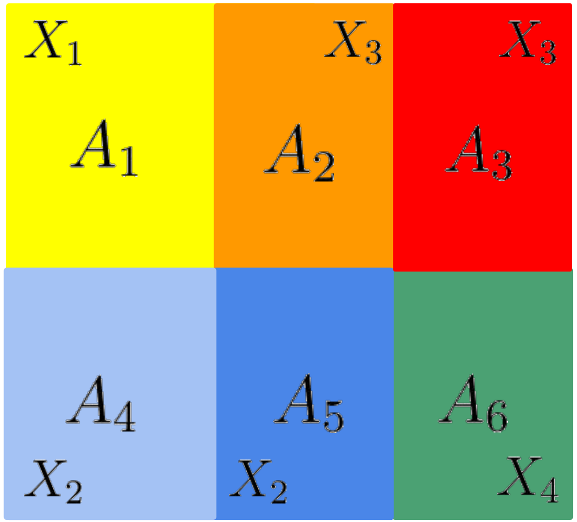
Let $\chi(f)$ denote the minimum number of rectangles in any monochromatic partition of $f^{-1}(1) \times f^{-1}(0)$

Columns labelled with $y \in f^{-1}(0)$

$$A = \sum_{i=1}^{\chi(f)} A_i$$

A

Rows labelled with $x \in f^{-1}(1)$



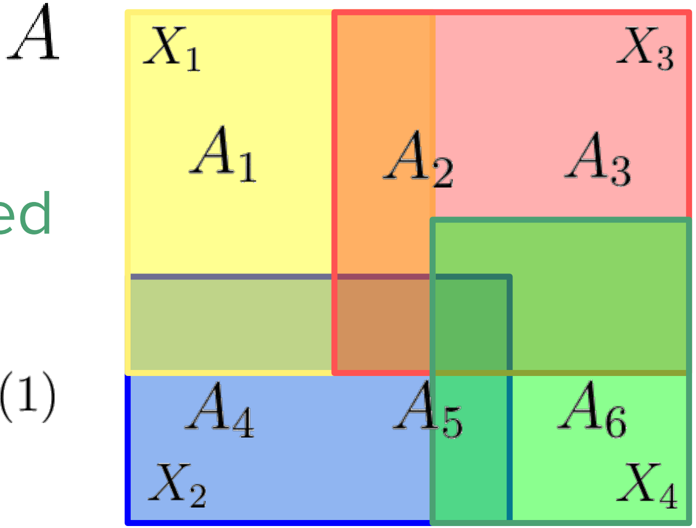
$$\operatorname{rank}(A) \leq \chi(f) \max_i \operatorname{rank}(A_i)$$

Let $\chi(f)$ denote the minimum number of rectangles in any monochromatic partition of $f^{-1}(1) \times f^{-1}(0)$

$$A = \sum_{i=1}^{\chi(f)} A_i$$

Columns labelled with $y \in f^{-1}(0)$

Rows labelled with $x \in f^{-1}(1)$



$$\mathrm{rank}(A) \le \chi(f) \max_i \mathrm{rank}(A_i)$$
$$\le \chi(f) \max_{i \in [n]} \mathrm{rank}(A \upharpoonright X_i)$$

Rearranging,

$$\chi(f) \ge \frac{\mathrm{rank}(A)}{\max_{i \in [n]} \mathrm{rank}(A \upharpoonright X_i)}$$

Rank Measure

Theorem [Razb. 90]. For any monotone boolean function f and any $f^{-1}(1) \times f^{-1}(0)$ matrix A over any field, the quantity

$$\mu_A(f) = \frac{\operatorname{rank}(A)}{\max_{i \in [n]} \operatorname{rank}(A \restriction X_i)}$$

is a lower bound on $\chi(f)$ (and the monotone formula size of f).

Theorem [G. 01, RPRC. 16]. $\mu_A(f)$ is also a lower bound on monotone switching networks, monotone span programs, and monotone comparator circuits computing f.

Rank Measure

Theorem [Razb. 90]. For any monotone boolean function f and any $f^{-1}(1) \times f^{-1}(0)$ matrix A over any field, the quantity

$$\mu_A(f) = \frac{\text{rank}(A)}{\max_{i \in [n]} \text{rank}(A \upharpoonright X_i)}$$

is a lower bound on $\chi(f)$ (and the monotone formula size of f).

Main Theorem (Restated). There is an explicit function f computable in NP and a matrix A such that $\mu_A(f) \geq 2^{\alpha n}$.

Proving Lower Bounds on $\mu_A(f)$

Theorem [Razb. 90] There is a monotone boolean function $f : \{0,1\}^n \to \{0,1\}$ in NP and a 0/1 matrix A satisfying

$$\mu_A(f) \geq n^{\Omega(\log n)}$$

[RPRC 16, PR 17] "Lifting theorem" to prove lower bounds against $\mu_A(f)$

1. Reduce lower bounds on $\mu_A(f)$ to **query complexity** lower bounds for a search problem $\mathrm{Search}(\mathcal{C})$ related to $\text{KW-Search}^+(f)$
2. Prove strong query complexity lower bounds for $\mathrm{Search}(\mathcal{C})$

Search Problems and Algebraic Gaps

$\mathcal{C} = C_1 \wedge C_2 \wedge \ldots \wedge C_m$ is an unsatisfiable **k**-CNF with variables **z**.

$\mathrm{Search}(\mathcal{C})$:= given assignment to **z**, output index of falsified clause.

Algebraic Gap Complexity

Ex. $$\mathcal{C} = \overline{x}_1 \wedge \overline{x}_2 \wedge \cdots \wedge \overline{x}_n \wedge \left(\bigvee_{i=1}^{n} x_i \right)$$

Algebraic Gap Complexity

Ex. $$\mathcal{C} = \overline{x}_1 \wedge \overline{x}_2 \wedge \cdots \wedge \overline{x}_n \wedge \left(\bigvee_{i=1}^{n} x_i \right)$$

Certificate = minimal partial restriction falsifying a clause

Algebraic Gap Complexity

Ex. $$\mathcal{C} = \overline{x}_1 \wedge \overline{x}_2 \wedge \cdots \wedge \overline{x}_n \wedge \left(\bigvee_{i=1}^{n} x_i \right)$$

$$\mathrm{Cert}(\mathcal{C}) \quad x_1 = 1 \;\; x_2 = 1 \quad\quad x_n = 1 \quad x_1 = 0, x_2 = 0, \cdots, x_n = 0$$

Certificate = minimal partial restriction falsifying a clause

Algebraic Gap Complexity

Ex. $\mathcal{C} = \overline{x}_1 \wedge \overline{x}_2 \wedge \cdots \wedge \overline{x}_n \wedge \left(\bigvee_{i=1}^{n} x_i \right)$

$\mathrm{Cert}(\mathcal{C})$ $\quad x_1 = 1 \quad x_2 = 1 \qquad x_n = 1 \qquad x_1 = 0, x_2 = 0, \cdots, x_n = 0$

Algebraic Gap Complexity. Find a polynomial $p : \{0,1\}^n \to \mathbb{R}$

so that $\mathrm{gap}_p(\mathcal{C}) = \deg(p) - \max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(p \restriction \pi)$ is maximized.

Algebraic Gap Complexity

Ex. $\mathcal{C} = \overline{x}_1 \wedge \overline{x}_2 \wedge \cdots \wedge \overline{x}_n \wedge \left(\bigvee_{i=1}^{n} x_i \right)$

$\mathrm{Cert}(\mathcal{C})$ $\quad x_1 = 1 \quad x_2 = 1 \quad x_n = 1 \quad x_1 = 0, x_2 = 0, \cdots, x_n = 0$

Algebraic Gap Complexity. Find a polynomial $p : \{0,1\}^n \to \mathbb{R}$ so that $\mathrm{gap}_p(\mathcal{C}) = \deg(p) - \max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(p \upharpoonright \pi)$ is maximized.

$p = OR_n \implies \deg(OR_n) = n$ and $\max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(OR_n) = 0$

Algebraic Gap Complexity vs. Rank Measure

Algebraic Gap Complexity. Given $\mathrm{Search}(\mathcal{C})$, find polynomial $p : \{0,1\}^n \to \mathbb{R}$ so that $\mathrm{gap}_p(\mathcal{C}) = \deg(p) - \max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(p \restriction \pi)$ is maximized.

Rank Measure $\mu_A(f)$. Given $f : \{0,1\}^N \to \{0,1\}$, find matrix A such that

$$\mu_A(f) = \frac{\mathrm{rank}(A)}{\max_{i \in [n]} \mathrm{rank}(A \restriction X_i)}$$

is maximized.

Rank Measure Lifting

Theorem [RPRC 16].

For any unsatisfiable **k**-CNF $\mathcal{C}$ with **m** clauses there is a function $f_{\mathcal{C}}$ computable in NP with $N \leq m^{2k+1}$ variables and a real matrix A such that

$$\mu_A(f_{\mathcal{C}}) \geq \Omega(m^{\mathrm{gap}(\mathcal{C})}) \geq \Omega(N^{\mathrm{gap}(\mathcal{C})/2k+1})$$

Rank Measure Lifting

Theorem [RPRC 16].

For any unsatisfiable **k**-CNF $\mathcal{C}$ with **m** clauses there is a function $f_{\mathcal{C}}$ computable in NP with $N \leq m^{2k+1}$ variables and a real matrix A such that

$$\mu_A(f_{\mathcal{C}}) \geq \Omega(m^{\mathrm{gap}(\mathcal{C})}) \geq \Omega(N^{\mathrm{gap}(\mathcal{C})/2k+1})$$

[RPRC 16]. $\mathcal{C}$ = “pebbling contradiction”, then $\mathrm{gap}(\mathcal{C}) \geq m/\log m \geq \tilde{\Omega}(N^{1/2k+1})$

Yields $2^{\Omega(N^{\varepsilon})}$ lower bounds!

Problem is the number of variables!

Gadget Size Blues

Query Complexity		$\leq$	**Circuit Complexity**	
Decision Tree Depth	[RM 99]		Monotone Circuit Depth	[RM 99]
Critical Block Sensitivity	[HN 12, GP 16]		Avg. Case Monotone Depth	[HN 12, GP 16]
Algebraic Gap Complexity	[RPRC 16]		**(Logarithm of) Rank Measure**	[RPRC 16]

For decision trees vs. depth, current constructions yield $N = \omega(m)$ variables.

For critical block sensitivity, we can take $N = O(m)$ variables, but best query lower bounds are $\Omega(m/\log m)$.

Rank Measure Lifting

Theorem [RPRC 16].

For any unsatisfiable **k**-CNF $\mathcal{C}$ with **m** clauses there is a function $f_{\mathcal{C}}$ computable in NP with $N \leq m^{2k+1}$ variables and a real matrix A such that

$$\mu_A(f_{\mathcal{C}}) \geq \Omega(m^{\text{gap}(\mathcal{C})}) \geq \Omega(N^{\text{gap}(\mathcal{C})/2k+1})$$

Rank Measure Lifting (Refined)

Theorem [PR 17].

For any unsatisfiable **O(1)**-CNF $\mathcal{C}$ with **m** clauses **satisfying** $\mathrm{gap}(\mathcal{C}) = \Omega(m)$ there is a function $f_{\mathcal{C}}$ computable in NP with $N = O(m)$ variables and a real matrix A such that

$$\mu_A(f_{\mathcal{C}}) \geq 2^{\Omega(m)} \geq 2^{\Omega(N)}$$

Rank Measure Lifting (Refined)

Theorem [PR 17].

For any unsatisfiable **O(1)**-CNF $\mathcal{C}$ with **m** clauses **satisfying** $\text{gap}(\mathcal{C}) = \Omega(m)$ there is a function $f_{\mathcal{C}}$ computable in NP with $N = O(m)$ variables and a real matrix A such that

$$\mu_A(f_{\mathcal{C}}) \geq 2^{\Omega(m)} \geq 2^{\Omega(N)}$$

Proof. [RPRC 16] $\text{KW-Search}^{+}(f_{\mathcal{C}}) \equiv \text{Search}(\mathcal{C} \circ g^n(x, y))$

Rank of **pattern matrix** $A = [p(g^n(x, y))]_{x,y \in \mathcal{X}^n \times \mathcal{Y}^n} \approx \exp(\deg(p))$

Proving Large Algebraic Gaps

Algebraic Gap Complexity. Given $\mathrm{Search}(\mathcal{C})$, find polynomial $p : \{0,1\}^n \to \mathbb{R}$ so that $\mathrm{gap}_p(\mathcal{C}) = \deg(p) - \max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(p \restriction \pi)$ is maximized.

Proving Large Algebraic Gaps

Algebraic Gap Complexity. Given $\mathrm{Search}(\mathcal{C})$, find polynomial $p : \{0,1\}^n \to \mathbb{R}$ so that $\mathrm{gap}_p(\mathcal{C}) = \deg(p) - \max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(p \upharpoonright \pi)$ is maximized.

Tseitin Principle. Let **G** be a k-regular graph with an odd number of vertices.

Proving Large Algebraic Gaps

Algebraic Gap Complexity. Given $\mathrm{Search}(\mathcal{C})$, find polynomial $p : \{0,1\}^n \to \mathbb{R}$ so that $\mathrm{gap}_p(\mathcal{C}) = \deg(p) - \max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(p \restriction \pi)$ is maximized.

Tseitin Principle. Let **G** be a k-regular graph with an odd number of vertices.

	Variables		**Constraints**	
$\mathrm{Tseitin}_G$	z_{uv}	$uv \in E$	$\bigoplus_{u \sim v} z_{uv} = 1$	$v \in V$

Proving Large Algebraic Gaps

Algebraic Gap Complexity. Given $\mathrm{Search}(\mathcal{C})$, find polynomial $p : \{0,1\}^n \to \mathbb{R}$ so that $\mathrm{gap}_p(\mathcal{C}) = \deg(p) - \max_{\pi \in \mathrm{Cert}(\mathcal{C})} \deg(p \restriction \pi)$ is maximized.

Tseitin Principle. Let **G** be a k-regular graph with an odd number of vertices.

	Variables		**Constraints**	
$\mathrm{Tseitin}_G$	z_{uv}	$uv \in E$	$\bigoplus_{u \sim v} z_{uv} = 1$	$v \in V$

Theorem. $\mathrm{gap}(\mathrm{Tseitin}_G) \geq \mathrm{Expansion}(G) \cdot m/3d$

Proof. Reduction to resolution width of $\mathrm{Tseitin}_G$

Rank Measure Lifting

Theorem [PR 17]. For any unsatisfiable **O(1)**-CNF $\mathcal{C}$ with **m** clauses **satisfying** $\text{gap}(\mathcal{C}) = \Omega(m)$ there is a function $f_\mathcal{C}$ computable in NP with $N = O(m)$ variables and a real matrix A such that

$$\mu_A(f_\mathcal{C}) \geq 2^{\Omega(m)} \geq 2^{\Omega(N)}$$

Theorem. $\text{gap}(\text{Tseitin}_G) \geq \text{Expansion}(G) \cdot m/3d$

Choose G to be a strong constant-degree expander and the main theorem is proved!

Conclusion

Prove the first strongly exponential lower bounds for any explicit function, asymptotically matching non-explicit lower bounds from counting in the monotone setting.

Can we sharpen it further?

Further applications of the framework? (In particular, a deeper understanding of the **algebraic gap complexity** and other exotic query complexity measures for search problems.)

Thanks!