Lower Bounds for Unrestricted Boolean Circuits: Open Problems

Alexander S. Kulikov

Steklov Institute of Mathematics at St. Petersburg

CSR 2018

June 09, 2018

Computing Boolean Functions

Computing a Boolean function

$$f(x_1, x_2, x_3)\colon \{0,1\}^3 \to \{0,1\}$$

Computing Boolean Functions

Computing a Boolean function

$$f(x_1, x_2, x_3) \colon \{0,1\}^3 \to \{0,1\}$$

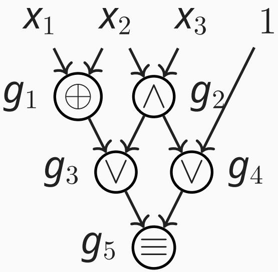
$$\begin{aligned}
g_1 &= x_1 \oplus x_2 \\
g_2 &= x_2 \wedge x_3 \\
g_3 &= g_1 \vee g_2 \\
g_4 &= g_2 \vee 1 \\
g_5 &= g_3 \equiv g_4
\end{aligned}$$

Computing Boolean Functions

Computing a Boolean function

$$f(x_1, x_2, x_3) \colon \{0,1\}^3 \to \{0,1\}$$

$$\begin{aligned}
g_1 &= x_1 \oplus x_2 \\
g_2 &= x_2 \wedge x_3 \\
g_3 &= g_1 \vee g_2 \\
g_4 &= g_2 \vee 1 \\
g_5 &= g_3 \equiv g_4
\end{aligned}$$



Fundamental Question

Given a Boolean function $f\colon \{0,1\}^n \to \{0,1\}$, what is the minimum number of gates needed to compute f?

Fundamental Question

Given a Boolean function $f\colon \{0,1\}^n \to \{0,1\}$, what is the minimum number of gates needed to compute f?

Does there exist an infinite sequence of functions $f_1, f_2, \ldots$ such that f_n has n inputs, $\bigcup_{n=1}^{\infty} f_n^{-1}(1) \in \mathrm{NP}$, and f_n requires superpoly(n) gates? (This would mean that $\mathrm{P} \neq \mathrm{NP}$)

Exponential Bounds

Lower Bound

Counting shows that almost all functions of n variables have circuit size $\Omega(2^n/n)$ [S49]

Upper Bound

Any function can be computed by circuits of size $(1 + o(1))2^n/n$ [L58]

Explicit Lower Bounds

The lower bound $\Omega(2^n/n)$ is non-constructive: it does not give an explicit function (i.e., a function from NP) with superpolynomial circuit size.

Explicit Lower Bounds

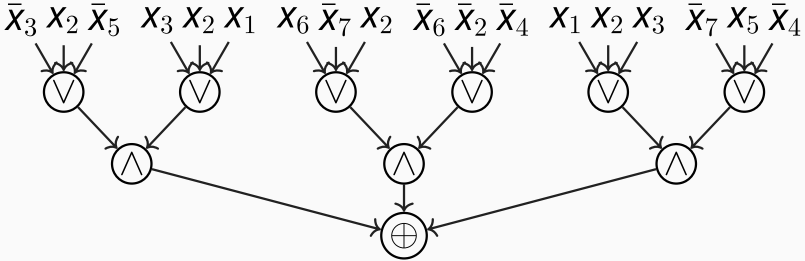
The lower bound $\Omega(2^n/n)$ is non-constructive: it does not give an explicit function (i.e., a function from NP) with superpolynomial circuit size.

What can we prove for explicit functions? What about restricted circuit classes?

Remainder of the Talk

- (Very brief) Overview of known lower bounds for restricted circuits
- (Brief) Overview of various approaches that could potentially lead to improved lower bounds for unrestricted circuits

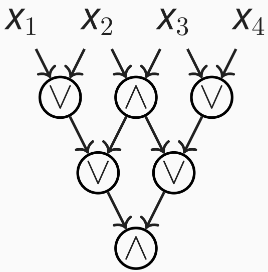
Restricted classes: constant depth circuits



- depth: constant, fan-in: unbounded
- exponential lower bounds: switching lemma [A83, FSS84, Y85, H86, R95], approximating polynomials [RS87]

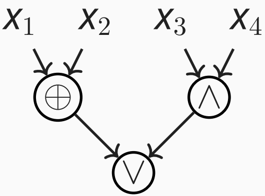
Restricted classes: monotone circuits

- fanin: 2
  fanout: unbounded
  operations: $\{\wedge, \vee\}$
- exponential lower bounds: method of approximations [R85, A85, AB87]



Restricted classes: formulas

- fanin: 2, fanout: 1
- n^2, n^3 lower bounds: random restrictions, universal functions, formal complexity measures [S61, N66, K71, A85, IN93, PZ93, H98]



$(x_1 \oplus x_2) \vee (x_3 \wedge x_4)$

Restricted vs Unrestricted

Restricted circuits

lower bounds:
n^3, $2^{n^{1/8}}$, $2^{n-o(n)}$

many beautiful techniques are known

Restricted vs Unrestricted

Restricted circuits

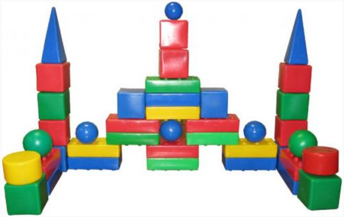
lower bounds:
n^3, $2^{n^{1/8}}$, $2^{n-o(n)}$

many beautiful techniques are known

Unrestricted circuits

lower bounds:
$2n$, $2.5n$, $3n$

just one simple technique is known



Quote

“This may seem quite depressing. It is.”

Saxena, Seshadhri, 2010. From Sylvester–Gallai Configurations to Rank Bounds: Improved Blackbox Identity Test for Depth-3 Circuits

Outline

1. Gate Elimination
2. Multi-Output Functions
3. Non-Gate-Elimination Lower Bounds
4. Symmetric Functions
5. Satisfiability Algorithms
6. Mass Production
7. Logarithmic Depth Circuits

Outline

1. Gate Elimination

   How to prove, say, a $3n$ lower bound for a Boolean function f?

2. Multi-Output Functions
3. Non-Gate-Elimination Lower Bounds
4. Symmetric Functions
5. Satisfiability Algorithms
6. Mass Production
7. Logarithmic Depth Circuits

Gate Elimination Method

- Show that f is resistant to about n substitutions

- Show that one can always find a substitution eliminating at least 3 gates

Lower Bounds

- The currently best known lower bound $\left(3+\frac{1}{86}\right)n$ is proved by gate elimination [FGHK16]
- The corresponding function f is affine disperser for sublinear dimension: f is non-constant on any affine subspace of $\{0,1\}^n$ of large enough dimension
- Explicit constructions of such functions were found relatively recently [BK12]

Linear Size Circuits for Affine Dispersers

All other functions used in lower bounds proofs ($2n$, $2.5n$, $3n$) have linear circuit size (at most $6n$)

Linear Size Circuits for Affine Dispersers

All other functions used in lower bounds proofs ($2n$, $2.5n$, $3n$) have linear circuit size (at most $6n$)

Open problem: Do there exist affine dispersers for sublinear dimension of linear circuit size?

Quadratic Dispersers

Open problem: Construct an explicit "quadratic" disperser f (even in NP, even with $o(n)$ outputs) that is not constant on any set $S \subseteq \{0,1\}^n$ of size at least $2^{n/100}$ that can be defined as

$$S = \{x \colon p_1(x) = \cdots = p_{2n}(x) = 0\}, \ \deg(p_i) \le 2.$$

Quadratic Dispersers

Open problem: Construct an explicit "quadratic" disperser *f* (even in NP, even with $o(n)$ outputs) that is not constant on any set $S \subseteq \{0,1\}^n$ of size at least $2^{n/100}$ that can be defined as

$$S = \{x \colon p_1(x) = \cdots = p_{2n}(x) = 0\}, \deg(p_i) \leq 2.$$

This will give an improved lower bound (about $3.1n$) [GK16]

Limitations of Gate Elimination

- Informally: Gate elimination proofs are tedious and usually consist of a long case analysis. It is difficult to imagine a relatively short gate elimination proof of, say, $4n$ lower bound

Limitations of Gate Elimination

- Informally: Gate elimination proofs are tedious and usually consist of a long case analysis. It is difficult to imagine a relatively short gate elimination proof of, say, $4n$ lower bound
- Formally, there exist circuits such that any substitution of the form $x \leftarrow g$, where g is an arbitrary function, removes no more than five gates from the circuit [GHkk16]. Therefore, one definitely needs new ideas to get something stronger than $5n$

Outline

1. Gate Elimination
2. Multi-Output Functions

   Can one prove stronger lower bounds for functions with multiple outputs?

3. Non-Gate-Elimination Lower Bounds
4. Symmetric Functions
5. Satisfiability Algorithms
6. Mass Production
7. Logarithmic Depth Circuits

Multi-Output Functions

- Computing several functions simultaneously is definitely not easier than computing any one of them

Multi-Output Functions

- Computing several functions simultaneously is definitely not easier than computing any one of them
- We do not know how to exploit this fact in lower bounds proofs: the strongest lower bound for functions with $o(n)$ outputs is the same as for functions with a single output

Multi-Output Functions

- Computing several functions simultaneously is definitely not easier than computing any one of them
- We do not know how to exploit this fact in lower bounds proofs: the strongest lower bound for functions with $o(n)$ outputs is the same as for functions with a single output
- For n outputs, the strongest lower bound is about $4n$ and follows from $3n$ lower bounds for single output functions

Multi-Output Functions

- Computing several functions simultaneously is definitely not easier than computing any one of them
- We do not know how to exploit this fact in lower bounds proofs: the strongest lower bound for functions with $o(n)$ outputs is the same as for functions with a single output
- For n outputs, the strongest lower bound is about $4n$ and follows from $3n$ lower bounds for single output functions

Open problem: How to prove a $5n$ lower bound for an n-to-n function?

Outline

1. Gate Elimination
2. Multi-Output Functions
3. Non-Gate-Elimination Lower Bounds

   Are there approaches other than gate elimination for proving lower bounds for unrestricted circuits?

4. Symmetric Functions
5. Satisfiability Algorithms
6. Mass Production
7. Logarithmic Depth Circuits

Other Lower Bounds

- Essentially, just a few and, alas, none of them is currently known to give a stronger than $2n$ lower bound

Other Lower Bounds

- Essentially, just a few and, alas, none of them is currently known to give a stronger than $2n$ lower bound
- $C(\mathrm{AND}, \mathrm{OR}) = 2n - 2$, idea: circuit reconstruction [BS84]

Other Lower Bounds

- Essentially, just a few and, alas, none of them is currently known to give a stronger than $2n$ lower bound
- $C(\mathrm{AND}, \mathrm{OR}) = 2n - 2$, idea: circuit reconstruction [BS84]
- $C(Ax) = 2n - o(n)$, idea: locating branching gates, wire counting [C94]

Other Lower Bounds

- Essentially, just a few and, alas, none of them is currently known to give a stronger than $2n$ lower bound
- $C(\mathrm{AND}, \mathrm{OR}) = 2n - 2$, idea: circuit reconstruction [BS84]
- $C(Ax) = 2n - o(n)$, idea: locating branching gates, wire counting [C94]

Open problem: Can any of these non-gate-elimination methods be extended to get stronger than $2n$ lower bounds?

Outline

1. Gate Elimination
2. Multi-Output Functions
3. Non-Gate-Elimination Lower Bounds
4. Symmetric Functions

   Can one prove a superlinear lower bound for a symmetric function?

5. Satisfiability Algorithms
6. Mass Production
7. Logarithmic Depth Circuits

Symmetric Functions

- While basic symmetric functions like parity, MOD_3, and majority are used to prove superpolynomial lower bounds in, e.g., constant depth circuit model, any symmetric function can be computed by a circuit of size $4.5n + o(n)$ [DKKY10]

Symmetric Functions

- While basic symmetric functions like parity, MOD_3, and majority are used to prove superpolynomial lower bounds in, e.g., constant depth circuit model, any symmetric function can be computed by a circuit of size $4.5n + o(n)$ [DKKY10]
- The function SUM_n is no easier than any symmetric function (with single output). It is known that $2.5n \leq C(\mathrm{SUM}_n) \leq 4.5n$

Symmetric Functions

- While basic symmetric functions like parity, MOD_3, and majority are used to prove superpolynomial lower bounds in, e.g., constant depth circuit model, any symmetric function can be computed by a circuit of size $4.5n + o(n)$ [DKKY10]
- The function SUM_n is no easier than any symmetric function (with single output). It is known that $2.5n \le C(\mathrm{SUM}_n) \le 4.5n$

Open problem: What is $C(\mathrm{SUM}_n)$?

Outline

1. Gate Elimination
2. Multi-Output Functions
3. Non-Gate-Elimination Lower Bounds
4. Symmetric Functions
5. Satisfiability Algorithms

   Given a circuit, how hard is it to find an assignment making this circuit to output 1?

6. Mass Production
7. Logarithmic Depth Circuits

Satisfiability Algorithms

- Faster than brute force search satisfiability algorithms imply circuit lower bounds [W11]

Satisfiability Algorithms

- Faster than brute force search satisfiability algorithms imply circuit lower bounds [W11]
- $O(2^n/n^{\omega(1)})$-time algorithm for checking satisfiability of circuits of size $2cn$ implies cn lower bounds (for a function with two outputs from E^{NP}) [JMV15]

Satisfiability Algorithms

- Faster than brute force search satisfiability algorithms imply circuit lower bounds [W11]
- $O(2^n/n^{\omega(1)})$-time algorithm for checking satisfiability of circuits of size $2cn$ implies cn lower bounds (for a function with two outputs from E^{NP}) [JMV15]
- We only know faster than brute force search algorithms for circuits of size at most $2.99n$ [GKST16]

Satisfiability Algorithms

- Faster than brute force search satisfiability algorithms imply circuit lower bounds [W11]
- $O(2^n/n^{\omega(1)})$-time algorithm for checking satisfiability of circuits of size $2cn$ implies cn lower bounds (for a function with two outputs from E^{NP}) [JMV15]
- We only know faster than brute force search algorithms for circuits of size at most $2.99n$ [GKST16]

Open problem: Do non-trivial satisfiability algorithms for circuits of size cn imply cn circuit lower bounds?

Outline

1. Gate Elimination
2. Multi-Output Functions
3. Non-Gate-Elimination Lower Bounds
4. Symmetric Functions
5. Satisfiability Algorithms
6. Mass Production

   Can one take a function of 20 bits of circuit size 100 and cook out of it a family of functions of circuit size $5n$*?*

7. Logarithmic Depth Circuits

Mass Production

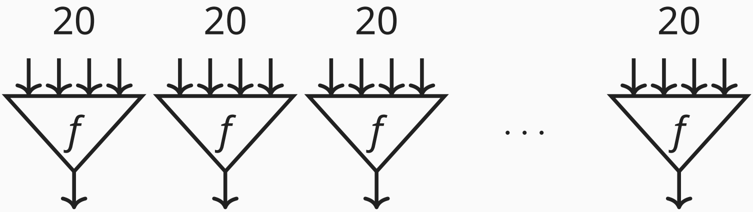
- Assume that $f\colon \{0,1\}^{20} \to \{0,1\}$ has circuit size 100

Mass Production

- Assume that $f\colon \{0,1\}^{20} \to \{0,1\}$ has circuit size 100
- Cook $g\colon \{0,1\}^{n} \to \{0,1\}^{n/20}$ out of it: g applies f to $n/20$ blocks of independent variables

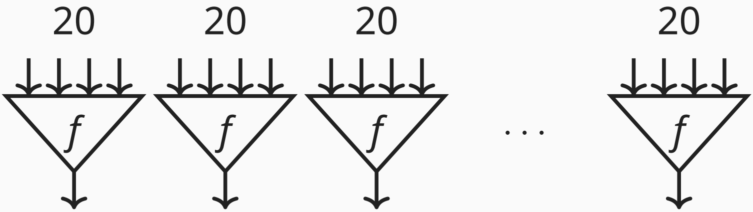
Mass Production

- Assume that $f\colon \{0,1\}^{20} \to \{0,1\}$ has circuit size 100
- Cook $g\colon \{0,1\}^{n} \to \{0,1\}^{n/20}$ out of it: g applies f to $n/20$ blocks of independent variables
- It is natural to expect that an optimal circuit for g looks as follows:



Mass Production

- Assume that $f\colon \{0,1\}^{20} \to \{0,1\}$ has circuit size 100
- Cook $g\colon \{0,1\}^{n} \to \{0,1\}^{n/20}$ out of it: g applies f to $n/20$ blocks of independent variables
- It is natural to expect that an optimal circuit for g looks as follows:



- But we don't know how to prove this!

Mass Production Effect

- We say that a mass production effect occurs when two copies of g can be computed by a circuit of size (much) smaller than $2C(g)$

Mass Production Effect

- We say that a mass production effect occurs when two copies of g can be computed by a circuit of size (much) smaller than $2C(g)$
- It is easy to show that it does not occur for very simple functions (say, when $C(g) = n - 1$)

Mass Production Effect

- We say that a mass production effect occurs when two copies of g can be computed by a circuit of size (much) smaller than $2C(g)$
- It is easy to show that it does not occur for very simple functions (say, when $C(g) = n - 1$)
- At the same time, it does occur for very hard functions: if $C(g) \approx 2^n/n$, then $C(g, g) \approx C(g)$

Mass Production Effect

- We say that a mass production effect occurs when two copies of g can be computed by a circuit of size (much) smaller than $2C(g)$
- It is easy to show that it does not occur for very simple functions (say, when $C(g) = n - 1$)
- At the same time, it does occur for very hard functions: if $C(g) \approx 2^n/n$, then $C(g, g) \approx C(g)$

Open problem: What are the functions avoiding mass production effect?

Outline

1. Gate Elimination
2. Multi-Output Functions
3. Non-Gate-Elimination Lower Bounds
4. Symmetric Functions
5. Satisfiability Algorithms
6. Mass Production
7. Logarithmic Depth Circuits

Can we at least prove superlinear lower bounds on circuits of logarithmic depth?

Logarithmic Depth Circuits

- Alas, currently, it is not known

Logarithmic Depth Circuits

- Alas, currently, it is not known
- However, if we further restrict the depth to be constant, then one can prove even superpolynomial lower bounds!

Logarithmic Depth Circuits

- Alas, currently, it is not known
- However, if we further restrict the depth to be constant, then one can prove even superpolynomial lower bounds!
- If a function can be computed by a circuit of logarithmic depth and linear size, then it can also be computed by an OR of CNF's of total size $2^{O(n/\log\log n)}$ [V83]

Logarithmic Depth Circuits

- Alas, currently, it is not known
- However, if we further restrict the depth to be constant, then one can prove even superpolynomial lower bounds!
- If a function can be computed by a circuit of logarithmic depth and linear size, then it can also be computed by an OR of CNF's of total size $2^{O(n/\log\log n)}$ [V83]

Open problem: Improve $2^{\sqrt{n}}$ lower bound for depth three circuits.

Constant Depth Circuits

- Lower bounds of the form $2^{n/k}$ are known for $\mathrm{OR} \circ \mathrm{AND} \circ \mathrm{OR}_k$ circuits (i.e., OR of k-CNFs) [PSZ97]

Constant Depth Circuits

- Lower bounds of the form $2^{n/k}$ are known for $\mathrm{OR} \circ \mathrm{AND} \circ \mathrm{OR}_k$ circuits (i.e., OR of k-CNFs) [PSZ97]

Open problem: Can one convert a circuit with s gates into a, say, $\mathrm{OR}_{2^{\frac{s}{4}}} \circ \mathrm{AND} \circ \mathrm{OR}_2$ formula?

Summary of Open Problems

1. Prove that there exists an affine disperser of linear circuit size!
2. Construct an explicit quadratic disperser!
3. Prove a $5n$ lower bound for an n-to-n function!
4. Prove $3n$ lower bound without gate elimination!
5. Find $C(\mathrm{SUM}_n)$!
6. Prove that faster than brute force SAT algorithm for circuits of size cn imply cn circuit lower bounds!
7. Construct functions avoiding mass production effect!
8. Convert lower bounds for depth-3 circuits to lower bounds for unrestricted circuits!

Thank you!