

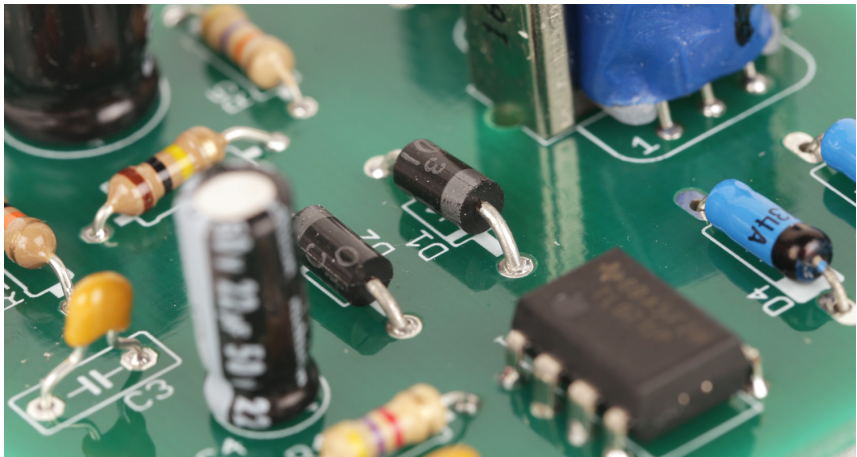
# GEMS OF TCS

## CIRCUIT COMPLEXITY

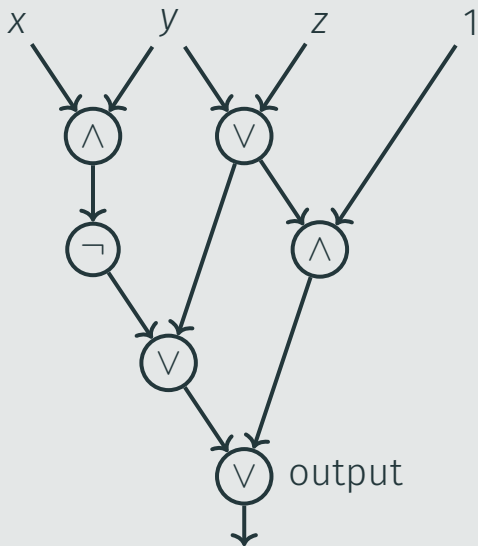
---

Sasha Golovnev

October 25, 2021



# Circuit



## Definition

A **circuit** is a directed acyclic graph of in-degree at most 2. Nodes of in-degree 0 are called **inputs** and are marked by Boolean variables and constants. Nodes of in-degree 1 and 2 are called **gates**: gates of in-degree 1 are labeled with NOT, gates of in-degree 2 are labeled with AND or OR. One of the sinks is marked as **output**.

# BOOLEAN CIRCUITS

$$f: \{0, 1\}^n \rightarrow \{0, 1\}$$

$$g_1 = \neg x_1$$

$$g_2 = x_2 \wedge x_3$$

$$g_3 = g_1 \vee g_2$$

$$g_4 = g_2 \vee 1$$

$$g_5 = g_3 \wedge g_4$$

# BOOLEAN CIRCUITS

$$f: \{0, 1\}^n \rightarrow \{0, 1\}$$

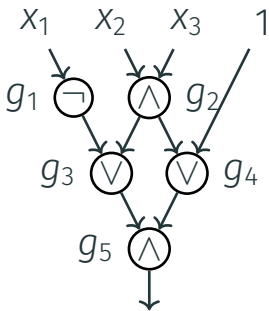
$$g_1 = \neg x_1$$

$$g_2 = x_2 \wedge x_3$$

$$g_3 = g_1 \vee g_2$$

$$g_4 = g_2 \vee 1$$

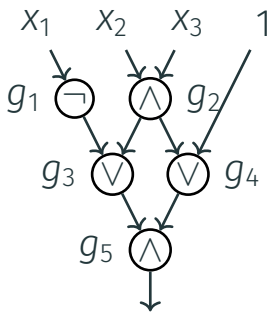
$$g_5 = g_3 \wedge g_4$$



# BOOLEAN CIRCUITS

$$f: \{0, 1\}^n \rightarrow \{0, 1\}$$

$$\begin{aligned}g_1 &= \neg X_1 \\g_2 &= X_2 \wedge X_3 \\g_3 &= g_1 \vee g_2 \\g_4 &= g_2 \vee 1 \\g_5 &= g_3 \wedge g_4\end{aligned}$$



Inputs:

$X_1, \dots, X_n, 0, 1$

Gates:

AND, OR, NOT

Fan-out:

unbounded

Depth:

unbounded

# EXPONENTIAL BOUNDS

## Lower Bound [Sha1949]

Almost all functions of  $n$  variables have circuit size

$$\geq 2^n / n$$



# EXPONENTIAL BOUNDS

## Lower Bound [Sha1949]

Almost all functions of  $n$  variables have circuit size

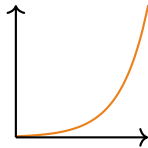
$$\geq 2^n/n$$

## Upper Bound [Lup1958]

Any function can be computed by a circuit of size

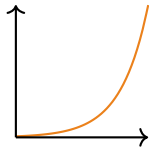
$$\leq 2^n/n$$

## EXPLICIT BOUNDS



Most functions have exponential circuit complexity

## EXPLICIT BOUNDS

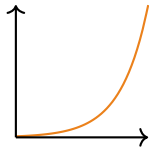


Most functions have **exponential** circuit complexity

**P**  $\neq$  **NP**

We want to prove **super-polynomial** lower bounds

## EXPLICIT BOUNDS

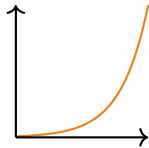


Most functions have **exponential** circuit complexity

**P**  $\neq$  **NP**

We want to prove **super-polynomial** lower bounds  
(for a function from **NP**)

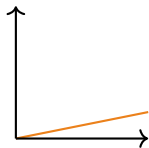
## EXPLICIT BOUNDS



Most functions have **exponential** circuit complexity

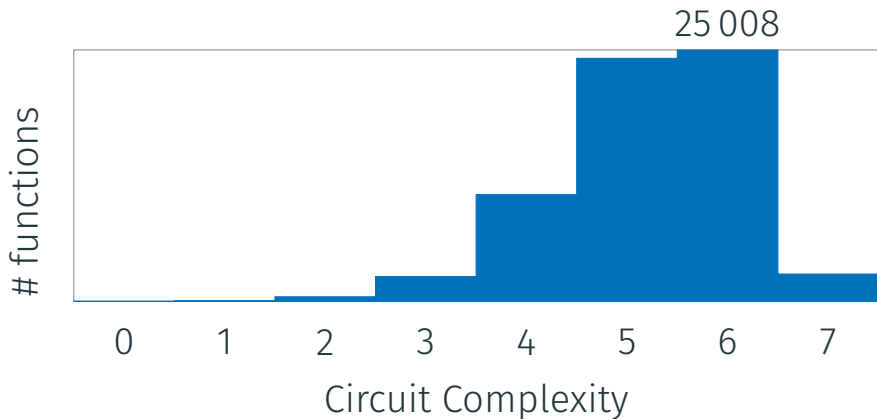
**P**  $\neq$  **NP**

We want to prove **super-polynomial** lower bounds  
(for a function from **NP**)

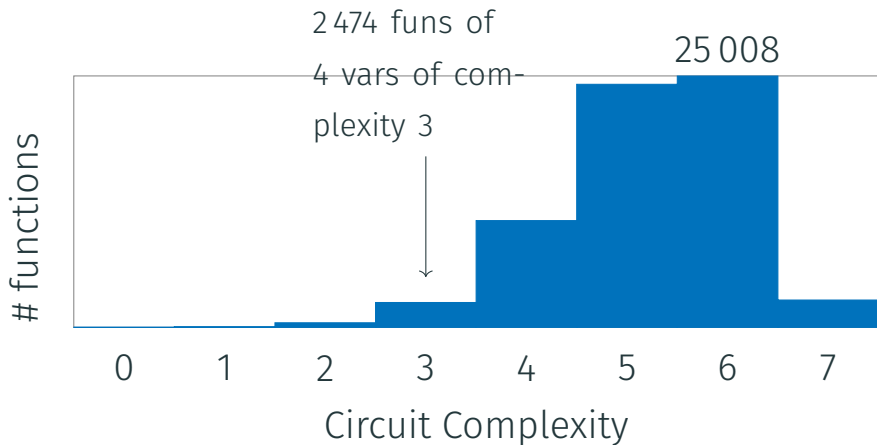


We can prove only  $\approx 5n$  lower bounds

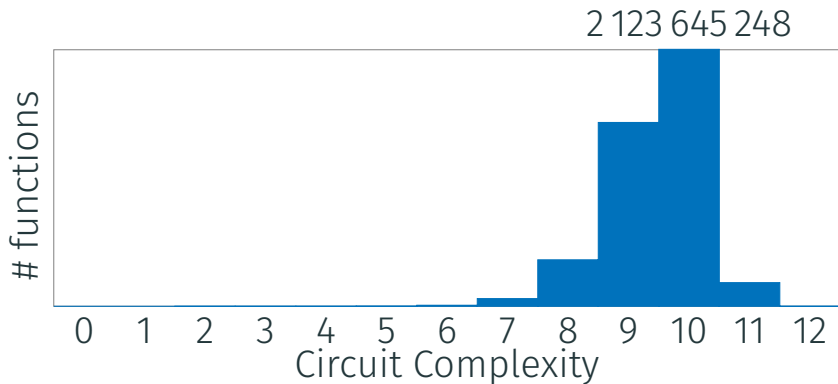
# CIRCUIT COMPLEXITY: $n = 4$



# CIRCUIT COMPLEXITY: $n = 4$

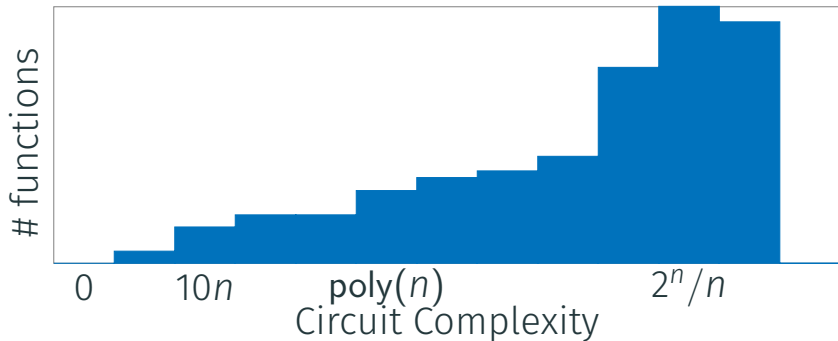


# CIRCUIT COMPLEXITY: $n = 5$

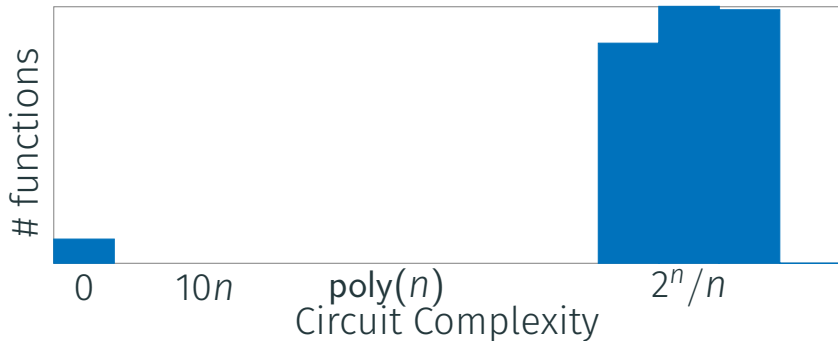




# CIRCUIT COMPLEXITY: GENERAL $n$



# CIRCUIT COMPLEXITY: GENERAL $n$



# HIERARCHY THEOREM

## Theorem

For any  $T \leq 2^n/n$ , there is a function  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  s.t.

$$\text{Size}(f) = T \pm n .$$

# HIERARCHY THEOREM

## Theorem

For any  $T \leq 2^n/n$ , there is a function  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  s.t.

$$\text{Size}(f) = T \pm n .$$

$$g_0(x) = 0, \forall x \in \{0, 1\}^n$$

# HIERARCHY THEOREM

## Theorem

For any  $T \leq 2^n/n$ , there is a function  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  s.t.

$$\text{Size}(f) = T \pm n .$$

$$g_0(x) = 0, \forall x \in \{0, 1\}^n$$

$$\text{Size}(g_0) = 1$$

# HIERARCHY THEOREM

## Theorem

For any  $T \leq 2^n/n$ , there is a function  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  s.t.

$$\text{Size}(f) = T \pm n.$$

$$g_0(x) = 0, \forall x \in \{0, 1\}^n \quad \text{Size}(h) \geq 2^n/n$$

$$\text{Size}(g_0) = 1$$

# HIERARCHY THEOREM

## Theorem

For any  $T \leq 2^n/n$ , there is a function  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  s.t.

$$\text{Size}(f) = T \pm n.$$

$$g_0(x) = 0, \forall x \in \{0, 1\}^n$$

$$\text{Size}(g_0) = 1$$

$$\text{Size}(h) \geq 2^n/n$$

$$h: \{0, 1\}^n \rightarrow \{0, 1\}$$

# HIERARCHY THEOREM

## Theorem

For any  $T \leq 2^n/n$ , there is a function  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  s.t.

$$\text{Size}(f) = T \pm n.$$

$$g_0(x) = 0, \forall x \in \{0, 1\}^n$$

$$\text{Size}(g_0) = 1$$

$$\text{Size}(h) \geq 2^n/n$$

$$h: \{0, 1\}^n \rightarrow \{0, 1\}$$

$$y_1, \dots, y_k \in \{0, 1\}^n$$

$$h(y_i) = 1$$



# HYBRID METHOD

$g_0(x) = 1$  never

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

$$g_3(x) = 1 \text{ if } x \in \{y_1, y_2, y_3\}$$

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

$$g_3(x) = 1 \text{ if } x \in \{y_1, y_2, y_3\}$$

...

$$g_k(x) = 1 \text{ if } x \in \{y_1, \dots, y_k\}$$

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

$$g_3(x) = 1 \text{ if } x \in \{y_1, y_2, y_3\}$$

...

$$h = g_k(x) = 1 \text{ if } x \in \{y_1, \dots, y_k\}$$

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

$$g_3(x) = 1 \text{ if } x \in \{y_1, y_2, y_3\}$$

...

$$h = g_k(x) = 1 \text{ if } x \in \{y_1, \dots, y_k\}$$

$$g_{i+1}(x) = g_i(x) \vee (x = y_{i+1})$$

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

$$g_3(x) = 1 \text{ if } x \in \{y_1, y_2, y_3\}$$

...

$$h = g_k(x) = 1 \text{ if } x \in \{y_1, \dots, y_k\}$$

$$g_{i+1}(x) = g_i(x) \vee (x = y_{i+1})$$

$$g_{i+1}(x) = g_i(x) \vee (x = 1011)$$



# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

$$g_3(x) = 1 \text{ if } x \in \{y_1, y_2, y_3\}$$

...

$$h = g_k(x) = 1 \text{ if } x \in \{y_1, \dots, y_k\}$$

$$g_{i+1}(x) = g_i(x) \vee (x = y_{i+1})$$

$$g_{i+1}(x) = g_i(x) \vee (x = 1011)$$

$$g_{i+1}(x) = g_i(x) \vee (x_1 \wedge \bar{x}_2 \wedge x_3 \wedge x_4)$$

# HYBRID METHOD

$$g_0(x) = 1 \text{ never}$$

$$g_1(x) = 1 \text{ if } x = y_1$$

$$g_2(x) = 1 \text{ if } x \in \{y_1, y_2\}$$

$$g_3(x) = 1 \text{ if } x \in \{y_1, y_2, y_3\}$$

...

$$h = g_k(x) = 1 \text{ if } x \in \{y_1, \dots, y_k\}$$

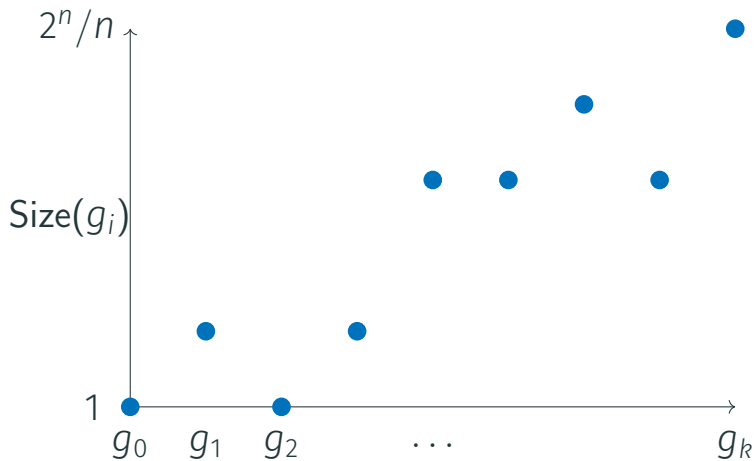
$$g_{i+1}(x) = g_i(x) \vee (x = y_{i+1})$$

$$g_{i+1}(x) = g_i(x) \vee (x = 1011)$$

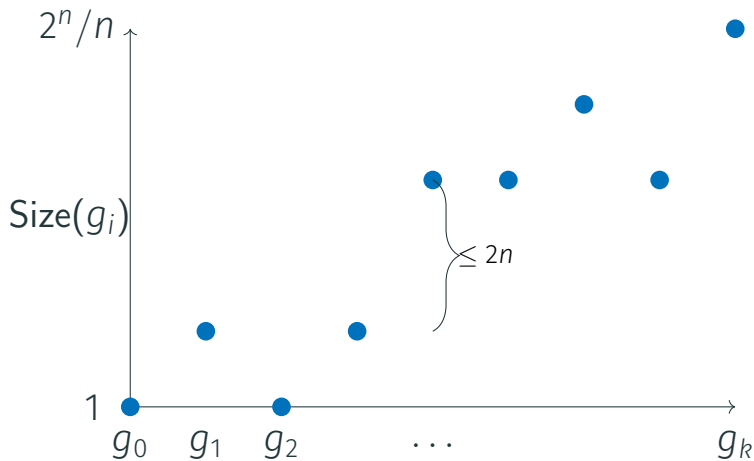
$$g_{i+1}(x) = g_i(x) \vee (x_1 \wedge \bar{x}_2 \wedge x_3 \wedge x_4)$$

$$\text{Size}(g_{i+1}) \leq \text{Size}(g_i) + 2n$$

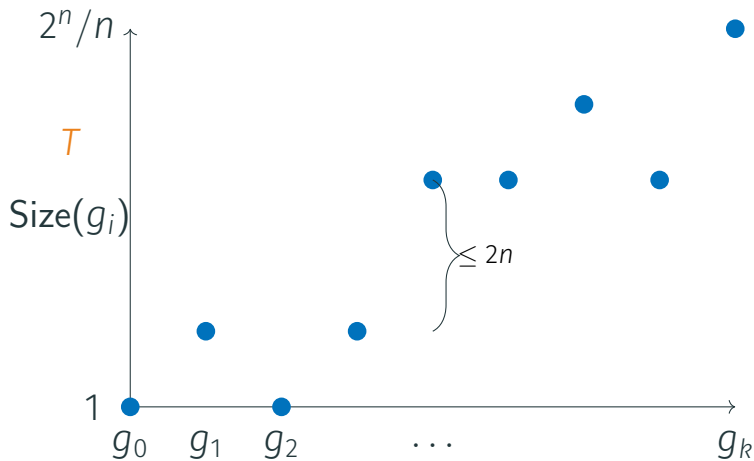
# HIERARCHY THEOREM



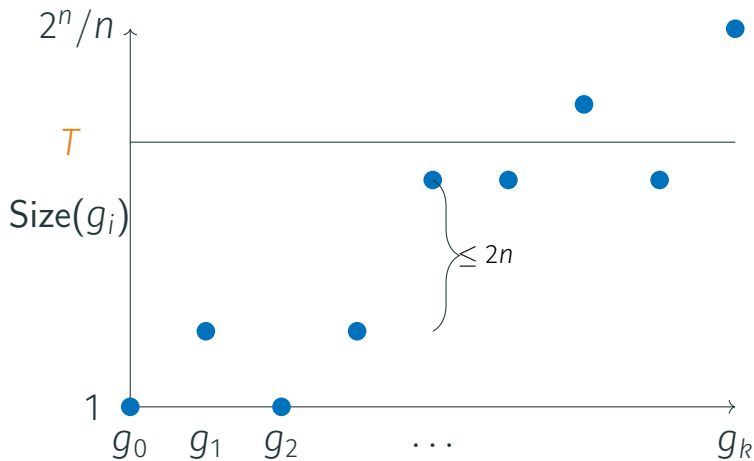
# HIERARCHY THEOREM



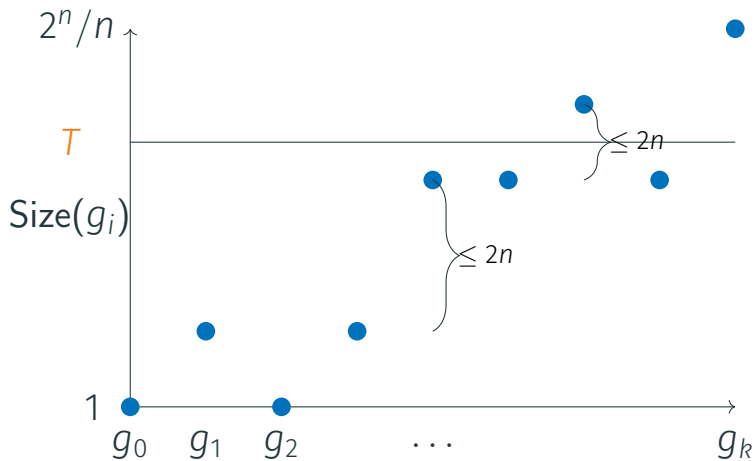
# HIERARCHY THEOREM



# HIERARCHY THEOREM



# HIERARCHY THEOREM



# HIERARCHY THEOREM

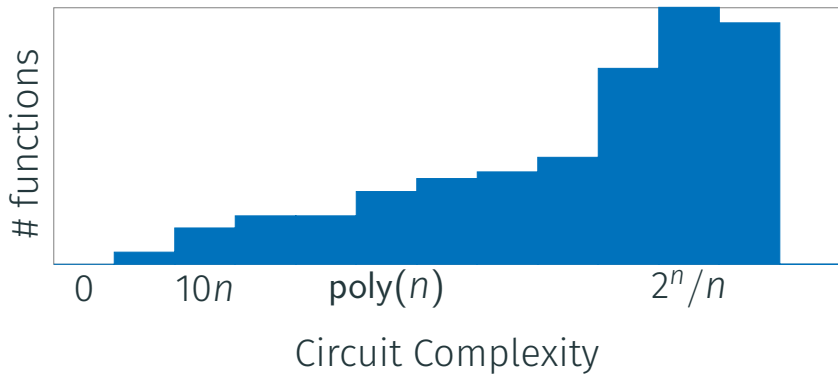
## Theorem

For any  $T \leq 2^n/n$ , there is a function  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  s.t.

$$\text{Size}(f) = T \pm n .$$

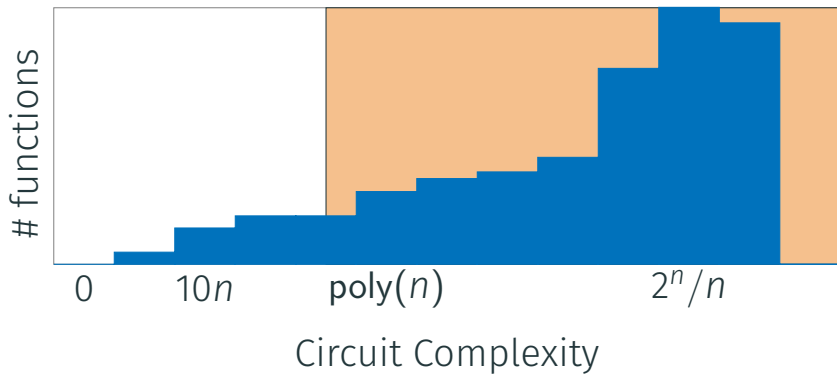


# GOAL



# GOAL

Find a hard function



# CIRCUIT COMPLEXITY

- Goal: Find a hard function

# CIRCUIT COMPLEXITY

- Goal: Find a hard function
- Lower bounds: what functions are hard

# CIRCUIT COMPLEXITY

- Goal: Find a hard function
- Lower bounds: what functions are hard
- Upper bounds: what functions are easy

# CIRCUIT UPPER BOUND. PROOF

## Upper Bound [Lup1958]

Any function can be computed by a circuit of size

$$\leq 10 \cdot 2^n$$

# CIRCUIT UPPER BOUND. PROOF

## Upper Bound [Lup1958]

Any function can be computed by a circuit of size

$$\leq 10 \cdot 2^n$$

$$f(x_1, \dots, x_n) = \begin{cases} f(1, x_2, \dots, x_n), & \text{if } x_1 = 1 \\ f(0, x_2, \dots, x_n), & \text{if } x_1 = 0 \end{cases}$$

# CIRCUIT UPPER BOUND. PROOF

## Upper Bound [Lup1958]

Any function can be computed by a circuit of size

$$\leq 10 \cdot 2^n$$

$$\begin{aligned} f(x_1, \dots, x_n) &= \begin{cases} f(1, x_2, \dots, x_n), & \text{if } x_1 = 1 \\ f(0, x_2, \dots, x_n), & \text{if } x_1 = 0 \end{cases} \\ &= (x_1 \wedge f(1, x_2, \dots, x_n)) \vee (\bar{x}_1 \wedge f(0, x_2, \dots, x_n)) \end{aligned}$$



# CIRCUIT UPPER BOUND. PROOF

## Upper Bound [Lup1958]

Any function can be computed by a circuit of size

$$\leq 10 \cdot 2^n$$

$$\begin{aligned} f(x_1, \dots, x_n) &= \begin{cases} f(1, x_2, \dots, x_n), & \text{if } x_1 = 1 \\ f(0, x_2, \dots, x_n), & \text{if } x_1 = 0 \end{cases} \\ &= (x_1 \wedge f(1, x_2, \dots, x_n)) \vee (\bar{x}_1 \wedge f(0, x_2, \dots, x_n)) \\ &= (x_1 \wedge g_1(x_2, \dots, x_n)) \vee (\bar{x}_1 \wedge g_0(x_2, \dots, x_n)) \end{aligned}$$

# CIRCUIT UPPER BOUND. PROOF

## Upper Bound [Lup1958]

Any function can be computed by a circuit of size

$$\leq 10 \cdot 2^n$$

$$f(x_1, \dots, x_n) = \begin{cases} f(1, x_2, \dots, x_n), & \text{if } x_1 = 1 \\ f(0, x_2, \dots, x_n), & \text{if } x_1 = 0 \end{cases}$$

$$= (x_1 \wedge f(1, x_2, \dots, x_n)) \vee (\bar{x}_1 \wedge f(0, x_2, \dots, x_n))$$

$$= (x_1 \wedge g_1(x_2, \dots, x_n)) \vee (\bar{x}_1 \wedge g_0(x_2, \dots, x_n))$$

$$\text{Size}(n) \leq 4 + 2 \text{Size}(n - 1) = O(2^n)$$

# CIRCUIT LOWER BOUND. PROOF

## Lower Bound [Sha1949]

Almost all functions of  $n$  variables have circuit size

$$\geq 2^n / (10n)$$