On the hardness of proving circuit lower bounds

Agrim Dewan,
Dept. of Computer Science and Automation,
Indian Institute of Science

Motivation

- One of the goals of complexity theory is to separate complexity classes like P vs NP, PSPACE vs EXP etc.

Motivation

- One of the goals of complexity theory is to separate complexity classes like P vs NP, PSPACE vs EXP etc.

- Early results used logical techniques like diagonalization to prove separation results.

Motivation

- One of the goals of complexity theory is to separate complexity classes like P vs NP, PSPACE vs EXP etc.

- Early results used logical techniques like diagonalization to prove separation results.

- Some examples are:
  - Time Hierarchy Theorem [Hartmannis-Stearns'1965] implying $P \subset EXP$.
  - Space Hierarchy Theorem [Stearns-Hartmannis-Lewis'1965] implying $P \neq SPACE(n)$ (it is not known if $P \not\subset SPACE(n)$ or $P \not\supset SPACE(n)$).
  - $NEXP^{NP} \not\subseteq P/poly$ [Kannan'1981] based on Shannon's counting argument.

Limitations of logical techniques

- Such techniques are abstract and general but limited in scope.

Limitations of logical techniques

- Such techniques are abstract and general but limited in scope.
- Unfortunately, these techniques hit the relativization barrier.

Limitations of logical techniques

- Such techniques are abstract and general but limited in scope.
- Unfortunately, these techniques hit the relativization barrier.
- There exist oracles A, B s.t. $P^A \neq NP^A$ and $P^B = NP^B$ [Baker-Gill-Solovay'1975].

Relativization barrier

- Main insight: diagonalization based proofs must hold w.r.t. oracles. Ironically, the proof uses diagonalization.

- Thus, any proof of P $\neq$ NP will have to be non-relativizing.

- [Wilson'1985] showed that there exists oracle A s.t. NEXP^{A} $\subseteq$ P^{A} /poly and every language in NP^{A} has linear sized circuits with A-oracle gates.

- Thus, proving NEXP $\nsubseteq$ P/poly also requires non-relativizing techniques.

Shift to circuit complexity

- Partly due to the relativization barrier, focus shifted to proving circuit lower bounds.

Shift to circuit complexity

- Partly due to the relativization barrier, focus shifted to proving circuit lower bounds.

- Circuit lower bounds imply lower bounds for algorithms, as circuits can be constructed from Turing Machines without much blowup.

Shift to circuit complexity

- Partly due to the relativization barrier, focus shifted to proving circuit lower bounds.

- Circuit lower bounds imply lower bounds for algorithms, as circuits can be constructed from Turing Machines without much blowup.

- The structured nature of circuits allows for combinatorial analysis.

Some circuit lower bounds

- Monotone circuits for Clique require $\geq n^{\Omega(\log n)}$ gates [Razborov’1985]. Improved to $\exp(\Omega(n/\log n)^{1/3})$ [Andreev’1985, Alon-Boppana’1987].

- [Furst-Saxe-Sipser’1981, Ajtai’1983] Parity is not in AC^0.

- [Razborov’1987, Smolensky’1987] For distinct primes p and q, MOD_q is not in $\mathrm{AC}^0[p]$.

Lack of general lower bounds

- Success in lower bounds for restricted classes, yet limited results for general circuits.

Lack of general lower bounds

- Success in lower bounds for restricted classes, yet limited results for general circuits.

- [Shannon'1948] showed almost all functions require exponential size circuits.

Lack of general lower bounds

- Success in lower bounds for restricted classes, yet limited results for general circuits.

- [Shannon’1948] showed almost all functions require exponential size circuits.

- Best known general circuit lower bounds:
  - There is an NP language with lower bound of $5n - o(n)$ [Iwama-Lachish-Morizumi-Raz’2005].
  - For every $k > 0$, PH contains languages with circuit complexity $\Omega(n^k)$[Kannan’1981].

Lack of general lower bounds

- Success in lower bounds for restricted classes, yet limited results for general circuits.
- [Shannon'1948] showed almost all functions require exponential size circuits.
- Best known general circuit lower bounds:
  - There is an NP language with lower bound of $5n - o(n)$ [Iwama-Lachish-Morizumi-Raz'2005].
  - For every $k > 0$, PH contains languages with circuit complexity $\Omega(n^k)$[Kannan'1981].
- A natural question arises:

Why is it hard to prove circuit lower bounds?

Lack of general lower bounds

- Success in lower bounds for restricted classes, yet limited results for general circuits.

- A natural question arises:

  Why is it hard to prove circuit lower bounds?

- [Razborov-Rudich’1994] gave a formal, complexity theoretic explanation for the lack of progress in proving circuit lower bounds which is called the *natural proofs barrier*.

Yet another barrier

- A lower bound proof strategy can be as follows:
  - Show that functions computable by *small size* circuits don't satisfy a certain property,
  - Then show that there is an explicit function with that property.

Yet another barrier

- A lower bound proof strategy can be as follows:
  - Show that functions computable by *small size* circuits don't satisfy a certain property,
  - Then show that there is an explicit function with that property.

- [Razborov-Rudich'1994] showed that if the property satisfies some criteria, which they call "natural", then the strategy won't suffice. Informally, they showed that
  - most of the known lower bounds proofs were "natural" in a sense and,
  - $P \neq NP$ does not have a "natural" proof assuming **pseudorandom functions** exist.

Natural proofs

- Let F_n be the set of all n-variate Boolean functions. Thus, $|F_n| = 2^{2^n}$.

Natural proofs

- Let F_n be the set of all n-variate Boolean functions. Thus, $|F_n| = 2^{2^n}$.

- Let Γ and Λ be classes of Boolean functions closed under fixing variables. For example, $\Gamma = \mathrm{P}$ and $\Lambda = \mathrm{P/poly}$.

Natural proofs

- Let F_n be the set of all n-variate Boolean functions. Thus, $|F_n| = 2^{2^n}$.

- Let Γ and Λ be classes of Boolean functions closed under fixing variables. For example, $\Gamma = \mathsf{P}$ and $\Lambda = \mathsf{P/poly}$.

- A property $\Phi: F_n \rightarrow \{0,1\}$ is said to be Γ-**natural** if the following hold:
  - **Constructivity**: $\Phi \in \Gamma$. Thus, for any $f \in F_n$, $\Phi(f)$ is computable by a circuit in Γ, where f is given as a truth table.
  - **Largeness**: $\Phi(f) = 1$ for at least $2^{-O(n)}$ fraction of the functions f in F_n.

Natural proofs

- Let F_n be the set of all n-variate Boolean functions. Thus, $|F_n| = 2^{2^n}$.
- Let Γ and Λ be classes of Boolean functions closed under fixing variables. For example, $\Gamma = \mathsf{P}$ and $\Lambda = \mathsf{P/poly}$.
- A property $\Phi: F_n \rightarrow \{0,1\}$ is said to be Γ-**natural** if the following hold:
  - **Constructivity**: $\Phi \in \Gamma$. Thus, for any $f \in F_n$, $\Phi\ (f)$ is computable by a circuit in Γ, where f is given as a truth table.
  - **Largeness**: $\Phi\ (f) = 1$ for at least $2^{-O(n)}$ fraction of the functions f in F_n.
- Further, Φ is **useful against** Λ if for any $f \in \Lambda$, $\Phi\ (f) = 0$.

Natural proofs

- $\Phi : F_n \rightarrow \{0,1\}$ is Γ-**natural**, if the following hold:
  - **Constructivity**: $\Phi \in \Gamma$, input is the *truth table*.
  - **Largeness**: $\Phi(f) = 1$ for at least $2^{-O(n)}$ fraction of the functions in F_n.

- Φ is **useful against** Λ if for any $f \in \Lambda$, $\Phi(f) = 0$.

- A lower bound proof against Λ which uses a property Φ that is Γ-**natural** and **useful against** Λ is called a natural proof.

Example: Parity $\notin$ AC^0

- Take Γ and Λ to be AC^0.

Example: Parity $\notin$ AC^0

- Take Γ and Λ to be AC^0.

- Main idea: Parity depends on *all* variables, while functions computable by AC^0 circuits can be made constant by fixing $\leq n - n^{\epsilon}$ variables.

Example: Parity $\notin$ AC^0

- Take Γ and Λ to be AC^0.

- Main idea: Parity depends on *all* variables, while functions computable by AC^0 circuits can be made constant by fixing $\leq n - n^{\epsilon}$ variables.

- The property Φ to be shown as AC^0-**natural** is

  $\Phi(f) = 1$ iff f cannot be made constant by fixing $\leq n - n^{\epsilon}$ variables.

  Note that Φ **is useful against** AC^0 by definition.

Example: Parity $\notin$ AC^0

$\Phi(f) = 1$ iff f cannot be made constant by fixing $\leq n - n^{\epsilon}$ variables.

- **Largeness:** The number of functions which become constant after fixing at most $n - k$ variables is at most

$$\binom{n}{k} 2^{2^{n-k}} \leq 2^{\frac{n}{2} + 2^{n-k}} < 2^{n 2^{n-k}} \ll 2^{2^n}.$$

Thus, $\Pr_{f \in_r F_n}[\Phi(f) = 1] \geq \frac{1}{2^{O(n)}}$.

Example: Parity $\notin$ AC^0

$\Phi(f) = 1$ iff f cannot be made constant by fixing $\leq n - n^{\epsilon}$ variables.

- **Constructivity:** Let S be an arbitrary $n-k$ sized subset of the variables.
- It is not hard to see that there is a depth-2 circuit, C_S, of size $2^{O(k)}$ s.t. $C_S(f) = 1$ iff f cannot be made constant by fixing the variables in S.
- By combining all $\binom{n}{k}$ circuits C_S, Φ can be tested by an AC^0 circuit of depth 3 and size $2^{O(n)}$, which is polynomial in the size of the truth table.

Pseudorandom function generator

- **Pseudorandom function generator** (**PRFG**): A Boolean function $f(x, y)$ in $n + m$ variables such that setting the y variables at random gives the n-variate subfunction $f_y(x)$.

Pseudorandom function generator

- **Pseudorandom function generator** (**PRFG**): A Boolean function $f(x, y)$ in $n + m$ variables such that setting the y variables at random gives the n-variate subfunction $f_y(x)$.

- Intuitively, the y-variables act as an index into a family of functions.

Pseudorandom function generator

- **Pseudorandom function generator** (**PRFG**): A Boolean function $f(x,y)$ in $n+m$ variables such that setting the y variables at random gives the n-variate subfunction $f_y(x)$.

- Intuitively, the y-variables act as an index into a family of functions.

- A PRFG f is **pseudorandom secure against** Γ if for any circuit $C \in \Gamma$,

$$\left| \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \right| < 2^{-n^2}.$$

Pseudorandom function generator

- **Pseudorandom function generator** (**PRFG**): A Boolean function $f(x, y)$ in $n + m$ variables such that setting the y variables at random gives the n-variate subfunction $f_y(x)$.

- Intuitively, the y-variables act as an index into a family of functions.

- A PRFG f is **pseudorandom secure against** Γ if for any circuit $C \in \Gamma$,

$$\left| \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \right| < 2^{-n^2}.$$

- Thus, no circuit from Γ can distinguish f_y from a truly random function.

Natural proofs barrier

- <u>**Theorem 1**</u>**:** If Λ contains a PRFG f secure against Γ, then there is no Γ-natural proof against Λ.

Natural proofs barrier

- **Theorem 1**: If Λ contains a PRFG f secure against Γ, then there is no Γ-natural proof against Λ.

- Alternatively, a natural proof not only proves a lower bound but also upper bounds the complexity of functions!!

Natural proofs barrier

- **Theorem 1:** If Λ contains a PRFG f secure against Γ, then there is no Γ-natural proof against Λ.

- Alternatively, a natural proof not only proves a lower bound but also upper bounds the complexity of functions!!

- Thus, a P-natural proof against P/poly implies P/poly does not contain PRFGs, contrary to the belief of their existence.

Proof of Theorem 1

$\Phi : F_n \to \{0,1\}$ is Γ**-natural,** if the following hold: 1) **Constructivity:** $\Phi \in \Gamma$**.** Input is the *truth table*. 2) **Largeness:** Φ holds for $\geq \frac{\lvert F_n \rvert}{2^{O(n)}}$ functions. Φ is **useful against** Λ if for any $f \in \Lambda$**,** $\Phi(f) = 0$**.** Assume Λ is P/poly and Γ is P**.**	A **PRFG** f is **pseudorandom secure against** Γ if for any circuit $C \in \Gamma$**,** $\left\lvert \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \right\rvert < 2^{-n^2}$.

- For contradiction, assume there exists a Γ-natural proof against Λ.

Proof of Theorem 1

$\Phi : F_n \to \{0,1\}$ is Γ**-natural,** if the following hold: 1) **Constructivity:** $\Phi \in \Gamma$. Input is the *truth table*. 2) **Largeness:** Φ holds for $\geq \frac{\lvert F_n \rvert}{2^{O(n)}}$ functions. Φ is **useful against** Λ if for any $f \in \Lambda$, $\Phi(f) = 0$. Assume Λ is P/poly and Γ is P.	A **PRFG** f is **pseudorandom secure against** Γ if for any circuit $C \in \Gamma$, $\left\lvert \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \right\rvert < 2^{-n^2}$.

- For contradiction, assume there exists a Γ-natural proof against Λ.
- Thus, there is a property Φ which is Γ-natural and useful against Λ.

Proof of Theorem 1

$\Phi : F_n \to \{0,1\}$ is Γ**-natural,** if the following hold: 1) **Constructivity:** $\Phi \in \Gamma$. Input is the *truth table*. 2) **Largeness:** Φ holds for $\geq \frac{\lvert F_n \rvert}{2^{O(n)}}$ functions. Φ is **useful against** Λ if for any $f \in \Lambda$, $\Phi(f) = 0$. Assume Λ is P/poly and Γ is P.	A **PRFG** f is **pseudorandom secure against** Γ if for any circuit $C \in \Gamma$, $\lvert \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \rvert < 2^{-n^2}$.

- For contradiction, assume there exists a Γ-natural proof against Λ.
- Thus, there is a property Φ which is Γ-natural and useful against Λ.
- As $f \in \Lambda$, $\Phi\big(f_y(x)\big) = 0$ for all $(x, y) \in \{0,1\}^{n+m}$ by **usefulness**.

Proof of Theorem 1

$\Phi : F_n \to \{0,1\}$ is Γ**-natural,** if the following hold: 1) **Constructivity:** $\Phi \in \Gamma$. Input is the *truth table*. 2) **Largeness:** Φ holds for $\geq \frac{\lvert F_n \rvert}{2^{O(n)}}$ functions. Φ is **useful against** Λ if for any $f \in \Lambda$, $\Phi(f) = 0$. Assume Λ is P/poly and Γ is P.	A **PRFG** f is **pseudorandom secure against** Γ if for any circuit $C \in \Gamma$, $\left\lvert \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \right\rvert < 2^{-n^2}$.

- **Largeness** implies $\Pr_{h \in_R F_n}[\Phi(h) = 1] \geq \frac{1}{2^{O(n)}}$.

Proof of Theorem 1

$\Phi : F_n \to \{0,1\}$ is Γ**-natural,** if the following hold: 1) **Constructivity:** $\Phi \in \Gamma$. Input is the *truth table*. 2) **Largeness:** Φ holds for $\geq \frac{\lvert F_n \rvert}{2^{O(n)}}$ functions. Φ is **useful against** Λ if for any $f \in \Lambda$, $\Phi(f) = 0$. Assume Λ is P/poly and Γ is P.	A **PRFG** f is **pseudorandom secure against** Γ if for any circuit $C \in \Gamma$, $\left\lvert \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \right\rvert < 2^{-n^2}$.

- By **constructivity**, there exists $C \in \Gamma$ s.t. $C(g) = \Phi(g)$ for any $g \in F_n$. This leads to the contradiction:

$$\left\lvert \Pr_{y \in_R \{0,1\}^m}[C(f_y(x)) = 1] - \Pr_{h \in_R F_n}[C(h) = 1] \right\rvert \geq \frac{1}{2^{O(n)}} \geq \frac{1}{2^{n^2}}.$$

On the existence of PRFGs

- PRFGs can be constructed from **Pseudorandom Generators** (**PRG**s) [Goldreich-Goldwasser-Micali'1984],

On the existence of PRFGs

- PRFGs can be constructed from **Pseudorandom Generators** (**PRG**s) [Goldreich-Goldwasser-Micali'1984],

- PRGs are efficiently computable functions which take as input a short random seed and stretch it to a longer "random-looking" string.

On the existence of PRFGs

- PRFGs can be constructed from **Pseudorandom Generators** (**PRG**s) [Goldreich-Goldwasser-Micali'1984],
- PRGs are efficiently computable functions which take as input a short random seed and stretch it to a longer "random-looking" string.
- PRGS are in turn constructible from **One-way functions** (**OWF**) [Hastad-Impagliazzo-Levin-Luby'1999].

On the existence of PRFGs

- PRFGs can be constructed from **Pseudorandom Generators** (**PRG**s) [Goldreich-Goldwasser-Micali'1984],

- PRGs are efficiently computable functions which take as input a short random seed and stretch it to a longer "random-looking" string.

- PRGS are in turn constructible from **One-way functions** (**OWF**) [Hastad-Impagliazzo-Levin-Luby'1999].

- OWFs are functions which are "easy to compute" but "hard to invert on the average".

On the existence of PRFGs

- **Conjecture 2**: There exists an OWF.

On the existence of PRFGs

- **<u>Conjecture 2</u>**: There exists an OWF.

- Some candidate OWFs:
  - Multiplication: Given $a, b \in \mathbb{Z}$, compute ab. The inverse is the factoring problem not known to be in P.
  - RSA function: Let $N \in \mathbb{Z}$, $\mathbb{Z}_N^* = \{a \in \mathbb{Z} |\ \mathrm{gcd}(a, N) = 1\}$ and $e \in \mathbb{Z}$ be co-prime to $|\mathbb{Z}_N^*|$. Given $a \in \mathbb{Z}_N^*$, compute a^e mod N. Factorization can be used to invert the function.

On the existence of PRFGs

- **<u>Conjecture 2</u>**: There exists an OWF.

- Some candidate OWFs:
  - Multiplication: Given $a, b \in \mathbb{Z}$, compute ab. The inverse is the factoring problem not known to be in P.
  - RSA function: Let $N \in \mathbb{Z}$, $\mathbb{Z}_N^* = \{a \in \mathbb{Z} | \gcd(a, N) = 1\}$ and $e \in \mathbb{Z}$ be co-prime to $|\mathbb{Z}_N^*|$. Given $a \in \mathbb{Z}_N^*$, compute a^e mod N. Factorization can be used to invert the function.

- Refer Chapter 9 of the Arora Barak textbook for a detailed discussion.

What makes “natural” proofs natural?

- A lower bound proof must identify a property not satisfied by small-size circuits. Thus, usefulness is necessary.

What makes "natural" proofs natural?

- A lower bound proof must identify a property not satisfied by small-size circuits. Thus, usefulness is necessary.

- [Razborov-Rudich'1994] gave a formal argument for largeness using *complexity measures*.

What makes “natural” proofs natural?

- A lower bound proof must identify a property not satisfied by small-size circuits. Thus, usefulness is necessary.

- [Razborov-Rudich’1994] gave a formal argument for largeness using *complexity measures*.

- The intuition is that if a specific function does not have size s circuits implies that random functions do not have size s circuits w.h.p.

Complexity measures

- A complexity measure $\mu: F_n \to \mathbb{Z}^{\geq 0}$ is a function such that

Complexity measures

- A complexity measure $\mu: F_n \to \mathbb{Z}^{\geq 0}$ is a function such that
  - **Small for trivial functions**: Let $\bar{x}$ be the complement of x. Then,

$$\mu(x) \leq 1 \text{ and } \mu(\bar{x}) \leq 1.$$

Complexity measures

- A complexity measure $\mu: F_n \to \mathbb{Z}^{\geq 0}$ is a function such that
  - **Small for trivial functions**: Let $\bar{x}$ be the complement of x. Then,

$$\mu(x) \leq 1 \text{ and } \mu(\bar{x}) \leq 1.$$

  - **Sub-additive**: For all functions f and g,

$$\mu(f \wedge g) \leq \mu(f) + \mu(g) \quad \text{and}$$

$$\mu(f \vee g) \leq \mu(f) + \mu(g).$$

Complexity measures

- For example, let $S(f)$ denote the smallest formula size for f.

- The function $\rho(f) = 1 + S(f)$ is a complexity measure (Easy exercise).

- More generally, if μ is a complexity measure, then $\mu(f)$ is a lower bound on $S(f)$.

Largeness via complexity measure

- **Lemma 3.** Let μ be a complexity measure. Suppose there exist a function f such that $\mu(f) \geq t$ for some number t. Then, for at least $\frac{1}{4}$ fraction of all functions $g \in F_n$, $\mu(g) \geq \frac{t}{4}$.

Largeness via complexity measure

- **<u>Lemma 3</u>.** Let μ be a complexity measure. Suppose there exist a function f such that $\mu(f) \geq t$ for some number t. Then, for at least $\frac{1}{4}$ fraction of all functions $g \in F_n$, $\mu(g) \geq \frac{t}{4}$.

- If a lower bound result uses a complexity measure, which also has the constructivity feature, then the measure is natural.

Proof of Lemma 3

- Suppose for contradiction, the conclusion is false.

Proof of Lemma 3

- Suppose for contradiction, the conclusion is false.
- Thus, $\mu(g) \geq \frac{t}{4}$ for $< \frac{1}{4}$-th fraction of functions $g \in F_n$.

Proof of Lemma 3

- Suppose for contradiction, the conclusion is false.
- Thus, $\mu(g) \geq \frac{t}{4}$ for $< \frac{1}{4}$-th fraction of functions $g \in F_n$.
- For $h \in_r F_n$, we can write $f = h \oplus g$, where $g = f \oplus h$.

Proof of Lemma 3

- Suppose for contradiction, the conclusion is false.
- Thus, $\mu(g) \geq \frac{t}{4}$ for $< \frac{1}{4}$-th fraction of functions $g \in F_n$.
- For $h \in_r F_n$, we can write $f = h \oplus g$, where $g = f \oplus h$.
- Observe that g is also a random function.

Proof of Lemma 3

- Now, $f = (\bar{g} \vee h) \wedge (g \vee \bar{h})$ implies

$$\mu(f) \leq \mu(\bar{g}) + \mu(h) + \mu(g) + \mu(\bar{h}).$$

Proof of Lemma 3

- Now, $f = (\bar{g} \vee h) \wedge (g \vee \bar{h})$ implies

$$\mu(f) \leq \mu(\bar{g}) + \mu(h) + \mu(g) + \mu(\bar{h}).$$

- As $\Pr_{h \in_r F_n}[\mu(h) \geq \frac{t}{4}] < \frac{1}{4}$ (also holds for $\bar{h}$, g and $\bar{g}$), by union bound, all four functions have measure $< \frac{t}{4}$ with non-zero probability.

Proof of Lemma 3

- Now, $f = (\bar{g} \vee h) \wedge (g \vee \bar{h})$ implies

$$\mu(f) \leq \mu(\bar{g}) + \mu(h) + \mu(g) + \mu(\bar{h}).$$

- As $\Pr_{h \in_r F_n}[\mu(h) \geq \frac{t}{4}] < \frac{1}{4}$ (also holds for $\bar{h}$, g and $\bar{g}$), by union bound, all four functions have measure $< \frac{t}{4}$ with non-zero probability.

- This implies $\mu(\bar{g}) + \mu(h) + \mu(g) + \mu(\bar{h}) < \frac{t}{4}$, a contradiction.

Why constructivity?

- Unlike largeness, [Razborov-Rudich’1994] gave no formal argument for constructivity.

Why constructivity?

- Unlike largeness, [Razborov-Rudich’1994] gave no formal argument for constructivity.

- They point out that known lower bound proofs use combinatorial properties satisfying constructivity.

Why constructivity?

- Unlike largeness, [Razborov-Rudich’1994] gave no formal argument for constructivity.

- They point out that known lower bound proofs use combinatorial properties satisfying constructivity.

- [Williams’2013] showed that proving NEXP $\not\subset \Gamma$ is equivalent to the existence of a constructive property against Γ.

Why constructivity?

- [Fan-Li-Yang'2021] initiated the study of *black-box natural properties*, with a stronger notion of constructivity which they call *black-box constructivity.*

Why constructivity?

- [Fan-Li-Yang’2021] initiated the study of *black-box natural properties*, with a stronger notion of constructivity which they call *black-box constructivity.*

- [Chen-Williams-Yang’2023] show the equivalence result of [Williams’2013] in the stronger notion as well, among other results.

Circumventing the Barrier

- Need to bypass largeness or constructivity.

Circumventing the Barrier

- Need to bypass largeness or constructivity.

- Arithmetization is a non-relativizing, non-natural technique used to prove IP = PSPACE.

Circumventing the Barrier

- Need to bypass largeness or constructivity.

- Arithmetization is a non-relativizing, non-natural technique used to prove $IP = PSPACE$.

- [Buhrman-Fortnow-Thierauf'1998] showed, using $IP = PSPACE$, that $MA_{EXP} \not\subset P/poly$, where MA_{EXP} is an exponential version of MA.

Circumventing the Barrier

- Need to bypass largeness or constructivity.
- Arithmetization is a non-relativizing, non-natural technique used to prove IP = PSPACE.
- [Buhrman-Fortnow-Thierauf’1998] showed, using IP = PSPACE, that $MA_{EXP} \not\subset P/poly$, where MA_{EXP} is an exponential version of MA.
- Unfortunately, arithmetization falls prey to a different barrier called *algebrization* [Aaronson-Wigderson’2008].

Circumventing the Barrier

- [Williams'2011] proved NEXP $\not\subset$ ACC by a combination of structural complexity and algorithmic ideas, where ACC $= \cup_p \mathrm{AC}^0[p]$.
- Proof technique bypasses the relativization, natural proofs and algebrization barrier.
- Later, [Murray-Williams'2018] showed that NQP $\not\subset$ ACC, where NQP is non-deterministic *quasi-polynomial* time.

Circumventing the Barrier

- Some other approaches include Algebraic Complexity Theory, Geometric Complexity Theory and Descriptive Complexity Theory.

- Refer Scott Aaronson's 2017 survey on P vs NP for a more detailed discussion.

Some perspectives

- Lance Fortnow and William Gasarch's blog https://blog.computationalcomplexity.org/2024/09/natural-proofs-is-not-barrier-you-think.html
- Richard Lipton's blog https://rjlipton.com/2009/03/25/whos-afraid-of-natural-proofs/
- Luca Trevisan's blog https://in-theory.blogspot.com/2006_07_30_archive.html