

New Lower Bounds and Derandomization for ACC, and a Derandomization-Centric View on the Algorithmic Method

Lijie Chen ✉

Miller Institute for Basic Research in Science at University of California, Berkeley, CA, USA

Abstract

In this paper, we obtain several new results on lower bounds and derandomization for ACC^0 circuits (constant-depth circuits consisting of AND/OR/MOD_m gates for a fixed constant m , a frontier class in circuit complexity):

1. We prove that any polynomial-time Merlin-Arthur proof system with an ACC^0 verifier (denoted by MA_{ACC^0}) can be simulated by a nondeterministic proof system with quasi-polynomial running time and polynomial proof length, on infinitely many input lengths. This improves the previous simulation by [Chen, Lyu, and Williams, FOCS 2020], which requires both quasi-polynomial running time and proof length.
2. We show that MA_{ACC^0} cannot be computed by fixed-polynomial-size ACC^0 circuits, and our hard languages are hard on a sufficiently dense set of input lengths.
3. We show that NEXP (nondeterministic exponential-time) does not have ACC^0 circuits of *sub-half-exponential* size, improving the previous *sub-third-exponential* size lower bound for NEXP against ACC^0 by [Williams, J. ACM 2014].

Combining our first and second results gives a conceptually simpler and derandomization-centric proof of the recent breakthrough result $\text{NQP} := \text{NTIME}[2^{\text{poly}(\log(n))}] \not\subseteq \text{ACC}^0$ by [Murray and Williams, SICOMP 2020]: Instead of going through an easy witness lemma as they did, we first prove an ACC^0 lower bound for a subclass of MA, and then derandomize that subclass into NQP, while retaining its hardness against ACC^0 .

Moreover, since our derandomization of MA_{ACC^0} achieves a polynomial proof length, we indeed prove that nondeterministic quasi-polynomial-time with $n^{\omega(1)}$ nondeterminism bits (denoted as $\text{NTIMEGUESS}[2^{\text{poly}(\log(n))}, n^{\omega(1)}]$) has no poly(n)-size ACC^0 circuits, giving a new proof of a result by Vyas. Combining with a win-win argument based on *randomized encodings* from [Chen and Ren, STOC 2020], we also prove that $\text{NTIMEGUESS}[2^{\text{poly}(\log(n))}, n^{\omega(1)}]$ cannot be $1/2 + 1/\text{poly}(n)$ -approximated by poly(n)-size ACC^0 circuits, improving the recent strongly average-case lower bounds for NQP against ACC^0 by [Chen and Ren, STOC 2020].

One interesting technical ingredient behind our second result is the construction of a PSPACE-complete language that is paddable, downward self-reducible, same-length checkable, and weakly error correctable. Moreover, all its reducibility properties have corresponding $\text{AC}^0[2]$ non-adaptive oracle circuits. Our construction builds and improves upon similar constructions from [Trevisan and Vadhan, Complexity 2007] and [Chen, FOCS 2019], which all require at least TC^0 oracle circuits for implementing these properties.

2012 ACM Subject Classification Theory of computation → Circuit complexity

Keywords and phrases Circuit Lower Bounds, Derandomization, Algorithmic Method, ACC

Digital Object Identifier 10.4230/LIPIcs.ITCS.2023.34

Related Version *Full Version*: <https://eccc.weizmann.ac.il/report/2022/183/>

Acknowledgements The author would like to thank Hanlin Ren, Roei Tell, Nikhil Vyas, and Ryan Williams for helpful discussions.



© Lijie Chen;

licensed under Creative Commons License CC-BY 4.0

14th Innovations in Theoretical Computer Science Conference (ITCS 2023).

Editor: Yael Tauman Kalai; Article No. 34; pp. 34:1–34:15

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Background. Proving unconditional circuit lower bounds for explicit functions (with the flagship problem of $\text{NP} \not\subseteq \text{P}_{/\text{poly}}$) is one of the holy grails in complexity theory and theoretical computer science. As the first step toward lower bounds for general circuits, constant-depth circuits received a lot of attention in the 1980s, with classical works culminating in exponential lower bounds against AC^0 [1, 12, 40, 15] (constant-depth circuits consisting of unbounded fan-in AND/OR gates) and $\text{AC}^0[q]$ (AC^0 circuits with MOD_q gates) for prime power q [25, 30].

Unfortunately, after the 1980s, the initial successes met an obstacle: lower bounds for $\text{AC}^0[m]$ have been extremely difficult to establish for composite m , despite the conjecture that $\text{AC}^0[m]$ cannot compute the majority function. In fact, it had been a notorious open question whether NEXP (nondeterministic exponential time) has polynomial-size ACC^0 circuits (ACC^0 denotes the union of $\text{AC}^0[m]$ for all constants m), until a decade ago Williams [36, 39] finally proved such a lower bound, via an *algorithmic* approach to circuit lower bounds [35, 37]. Combining many classical results from complexity theory, such as the nondeterministic time hierarchy theorem [27, 41], hardness vs. randomness [24], and the PCP theorem [3, 4], Williams’ work shows how non-trivial circuit-analysis algorithms can be generically applied to prove circuit lower bounds.

In 2018, Murray and Williams [22] significantly improved Williams’ lower bound by showing $\text{NQP} := \text{NTIME}[2^{\text{poly} \log(n)}]$ is not contained in ACC^0 . A line of recent work [11, 10, 6, 8] generalized the algorithmic approach to the average-case setting, culminating in the result that NQP cannot be $(1/2 + 1/\text{poly}(n))$ -approximated by $\text{poly}(n)$ -size ACC^0 circuits [8].¹

Motivation: making the algorithmic method direct. Most of the proofs following the algorithmic method are indirect and make heavy use of *proof by contradiction* or *win-win analysis*.² These proofs are considered by some to be conceptually hard to understand. A natural question is whether we can give a more direct (*i.e.*, no win-win analysis or proof by contradiction) proof of the separation $\text{NQP} \not\subseteq \text{ACC}^0$ [23], which might be (hopefully) easier to understand.

Inspired by [38, 33] and motivated by the goal of proving average-case lower bounds for NQP against ACC^0 , Chen [6] proposed a derandomization-centric perspective of the algorithmic method, which consists of the following two steps:

1. Assuming that the desired lower bound is false (*e.g.*, $\text{NQP} \subseteq \text{ACC}^0$ or NQP is average-case easy for ACC^0), [6] gave a derandomization of a certain sub-class of MA into NQP .
2. [6] then proved that this sub-class of MA contains a language that is hard against ACC^0 , and its hardness is retained after the aforementioned derandomization into NQP , from which the desired lower bound for NQP against ACC^0 follows immediately.

Following this perspective from [6], [9] managed to refine both of the two steps above to prove the strongly average-case lower bounds for NQP against ACC^0 . Still, the proofs from [6] and [9] are not direct, since the first step involves a proof by contradiction.³ Later, [7] managed to give a derandomization of the sub-class of MA with ACC^0 verifier (*i.e.*, MA_{ACC^0} ; see the full version for a formal definition.)

¹ The journal versions of [22] and [8] are [23] and [9], respectively.

² A win-win analysis often goes as follows: for some classes \mathcal{A} and \mathcal{B} , if $\mathcal{A} \subseteq \mathcal{B}$, then our desired lower bound follows in one way, and if $\mathcal{A} \not\subseteq \mathcal{B}$, then our desired lower bound follows in another way.

³ It can be interpreted as a “conditional derandomization”, since we can derandomize MA assuming the lower bound is false.

One may hope that we can replace the derandomization in the first step of [6] by a direct application of the new derandomization by [7], but this does not work for the following technical reason: The specific sub-class of MA considered in [6] is MA_{NC} , meaning that the verifier of the Merlin-Arthur protocol has $\text{polylog}(n)$ -depth circuits. This is (much) stronger than the class MA_{ACC^0} that [7] can derandomize. So to make [6]’s derandomization-centric perspective completely direct, we will need a lower bound for MA_{ACC^0} against ACC^0 , so that the derandomization of [7] can be applied to the corresponding hard language.

1.1 Our Results

Let \mathcal{C} be a circuit class. We use $\text{MA}_{\mathcal{C}}$ to denote the sub-class of MA whose verifier can be implemented by \mathcal{C} circuits. More formally, we say that $L \in \text{MA}$ if there is a polynomial-time algorithm $V(x, y, z)$ with $|x| = n$ and $|y|, |z| \leq \text{poly}(n)$ such that (1) $x \in L$ implies that there exists y satisfying $\Pr_z[V(x, y, z) = 1] = 1$ and (2) $x \notin L$ implies that for all y it holds that $\Pr_z[V(x, y, z) = 1] \leq 1/3$. And we say that $L \in \text{MA}_{\mathcal{C}}$ if V can be computed by polynomial-size \mathcal{C} circuits.⁴

1.1.1 A direct proof of $\text{NQP} \not\subseteq \text{ACC}^0$

Our first result is an affirmative answer to the question above by proving the following results.

► **Theorem 1 (Informal).** *For every $k, d_*, m_* \in \mathbb{N}$, there is a language $L \in \text{MA}_{\text{ACC}^0}$ and a constant $c \in \mathbb{N}$ such that for every sufficiently large $n \in \mathbb{N}$, there exists an input $m \in [n, n^c]$ such that L_m (the restriction of L on m -bit inputs) does not have m^k -size $\text{AC}_{d_*}^0[m_*]$ circuits.⁵*

Intuitively speaking, our hard language L is not only hard against m^k -size $\text{AC}_{d_*}^0[m_*]$ circuits, but its hard input lengths are not sparse in the sense that every interval $[n, n^c]$ contains a hard input length.⁶ This stronger hardness condition is crucial for the hard language to remain hard after the infinitely often derandomization from [7]. Combining Theorem 1 and the derandomization from [7], we then have an alternative proof of $\text{NQP} \not\subseteq \text{ACC}^0$.⁷

A subtle caveat. Interestingly, the direct derandomization proof above indeed only proves

$$(L1) : \text{NQP} \not\subseteq \text{AC}_{d_*}^0[m_*] \text{ for every } d_*, m_* \in \mathbb{N},$$

which is different from

$$(L2) : \text{NQP} \not\subseteq \text{ACC}^0.⁸$$

This issue occurs in [23] as well, as a direct application of their easy witness lemma (which is the core technical result of [23]) also only proves (L1). Nonetheless, [23] resolved this issue

⁴ Our formal definition of $\text{MA}_{\mathcal{C}}$ is slightly more technical and contains more languages (see the full version for details), but for the sake of the introduction, it might be easier to think of this simpler definition.

⁵ The hard language also needs one bit of advice per input length, we omit this technical issue in the introduction. We also indeed prove a weakly average-case lower bound against $\text{AC}_{d_*}^0[m_*]$; see the full version for more detail.

⁶ This is weaker than the “almost almost-everywhere” hardness notion in [23].

⁷ We remark that there are previous papers [20, 5] that simplifies some parts of Williams’ original proof [36]. However, these works do not change the high-level structure of Williams’ proof: they still argue by a proof of contradiction using an easy witness lemma. Our goal here is to eliminate the proof by contradiction completely.

by proving that (L1) implies (L2) via a simple win-win analysis.⁹ Unfortunately, we do not know how to get rid of this particular win-win analysis and leave this as an intriguing open question. Still, we believe that a direct proof of (L1) is already sufficiently interesting.

In Section 2, we give a *detailed overview* of new derandomization-centric alternative proofs for the main results of both [23] and [9].

1.1.2 An improved derandomization of MA_{ACC^0}

Our next result is an improvement of the derandomization of MA_{ACC^0} from [7]. Below, $\text{NTIMEGUESS}[T(n), G(n)]$ denotes the class of languages computable by nondeterministic algorithms with $T(n)$ time and $G(n)$ bits of nondeterminism.

► **Theorem 2.**

$$\text{MA}_{\text{ACC}^0} \subset i.o.\text{-NTIMEGUESS}[2^{\text{polylog}(n)}, \text{poly}(n)].$$

Theorem 2 is most interesting when viewed as a derandomization of randomized proof systems. It says that we can derandomize Merlin Arthur proof systems with ACC^0 verifier into a nondeterministic proof system with roughly the same proof length, albeit the running time goes up to a quasi-polynomial of the original running time. We hope such derandomization might have some applications in the derandomization of some recently purposed algebraic proof systems whose verifiers are randomized (*i.e.*, they are MA proof systems), for example [14].

As a consequence of Theorem 2, we give another proof of a result by [32].

► **Corollary 3** ([32]). *For every $\alpha(n) \geq \omega(1)$,*

$$\text{NTIMEGUESS}[2^{\text{polylog}(n)}, n^{\alpha(n)}] \not\subset \text{ACC}^0.$$

Combining the win-win argument from [9], we also strengthen Corollary 3 to the average-case, thus improving on [9].

► **Theorem 4.** *There is a $\beta \in \mathbb{N}$ such that for every $\alpha(n) \geq \omega(1)$, $\text{NTIMEGUESS}[2^{\log^\beta n}, n^{\alpha(n)}]$ cannot be $1/2 + 1/\text{poly}(n)$ -approximated by ACC^0 .*

Indeed, similar to all previous works following the algorithmic method, we prove a generic connection between circuit analysis algorithm and $\text{NTIMEGUESS}[2^{\text{polylog}(n)}, n^{\omega(1)}]$ lower bounds; see the full version for formal statements.

1.1.3 An improved lower bound for NEXP against ACC^0

Our third result is an improvement over the original lower bound for NEXP against ACC^0 in [39]. Roughly speaking, we say that a reasonable time-bound¹⁰ function $g(n)$ is *sub-half-exponential* if for every $k \in \mathbb{N}$, $g(g(n)^k)^k \leq 2^{n^{o(1)}}$, and we say call g *sub-third-exponential* if $g(g(g(n)^k)^k)^k \leq 2^{n^{o(1)}}$ for every $k \in \mathbb{N}$. In [39], it was proved that NEXP has no sub-third-exponential-size ACC^0 circuits. We improve that size bound to be sub-half-exponential.

⁹ If $\text{P} \not\subset \text{ACC}^0$, clearly (L2) is true. If $\text{P} \subset \text{ACC}^0$, then $\text{P}_{/\text{poly}}$ collapsed to $\text{AC}_{d_*}^0[m_*]$ for some fixed $d_*, m_* \in \mathbb{N}_{\geq 1}$, hence (L1) implies $\text{NQP} \not\subset \text{P}_{/\text{poly}}$.

¹⁰ see the full version for a formal definition.

► **Theorem 5.** *For every sub-half-exponential reasonable time-bound function $g(n)$, NE has no $g(n)$ -size ACC^0 circuits.*¹¹

Although our proof of Theorem 5 is still an indirect argument via a win-win analysis that is similar to and inspired by [33], we managed to nicely abstract out the indirect part by the following lemma. Recall that $\text{E} := \text{TIME}[2^{O(n)}]$ denotes the class of languages computable by single exponential-time deterministic algorithms.

► **Lemma 6 (Informal).** *Let $g(n)$ be a sub-half-exponential reasonable time-bound function. Assume that E has $g(n)$ -size ACC^0 circuits. Then*

$$\text{MATIME}_{\text{ACC}^0}[2^{n^{o(1)}}] \not\subseteq \text{i.o.-SIZE}[g(n)].$$

Theorem 5 then follows immediately by the following win-win argument: If E has no $g(n)$ -size ACC^0 , then we are done; otherwise, we apply our quasi-polynomial-time derandomization to show

$$\text{MATIME}_{\text{ACC}^0}[2^{n^{o(1)}}] \subseteq \text{i.o.-NTIME}[2^n],$$

which immediately implies that $\text{NTIME}[2^n] \not\subseteq \text{SIZE}[g(n)]$.

1.1.4 An improved construction of PSPACE-complete language

Finally, as a key technical ingredient behind the proof of Theorem 1, we construct a PSPACE-complete language with several nice reducibility properties that can all be implemented by non-adaptive $\text{AC}^0[2]$ circuits; see the full version for formal definitions of these properties.

► **Theorem 7.** *There is a PSPACE-complete language L^{PSPACE} that is paddable, non-adaptive $\text{AC}^0[2]$ downward self-reducible, non-adaptive $\text{AC}^0[2]$ same-length checkable and non-adaptive $\text{AC}^0[2]$ weakly error correctable.*

Theorem 7 improved upon a similar construction from [6, 9] (following [31, 26]), which gave a PSPACE-complete language that is paddable, non-adaptive TC^0 downward self-reducible, non-adaptive TC^0 same-length checkable and non-adaptive TC^0 weakly error correctable.

1.2 Intuition

Let us briefly discuss the intuition behind our results.

Lower bounds for MA_{ACC^0} and construction of PSPACE-complete languages. We will first discuss the intuition and technical challenges behind the proof of Theorem 1 and Theorem 7. Our proof of Theorem 1 follows the proof of a similar statement from [6].¹² In the following, we will ignore all minor details and only focus on the part relevant to us. In the framework of [6], to prove a lower bound against \mathcal{C} circuits, the verifier of the constructed hard MA language first guesses a \mathcal{C} circuit C of a certain size and then runs the instance-checker of the PSPACE-complete language from [6]. Thus, because the PSPACE-complete language from [6] only admits a TC^0 instance checker, the complexity of the verifier above is at least TC^0 , even if we only aim to prove lower bounds against weaker classes such as ACC^0 .

¹¹ Here $\text{NE} := \text{NTIME}[2^{O(n)}]$.

¹² Roughly speaking, [6] proved that $\text{MATIME}_{\text{NC}}[2^{\text{polylog}(n)}]$ has no $\log^d n$ -depth circuits for any fixed d .

Since our new Theorem 7 improves the instance-checker complexity of the PSPACE-complete language to $\text{AC}^0[2]$, we managed to prove Theorem 1 by plugging this new ingredient into the old framework of [6]. Along the way, we need several clever technical manipulations.

Next, we explain the ideas and technical challenges behind our proof of Theorem 7. It is instructive to recap some relevant details from the construction of the PSPACE-complete language L^{Che19} from [6]. Roughly speaking, on input length n , L^{Che19} computes a polynomial $P_n: \mathbb{F}_n^{m(n)} \rightarrow \mathbb{F}_n$, where $\mathbb{F}_n = \text{GF}(2^{r(n)})$ for some $r(n) \in \mathbb{N}_{\geq 1}$. Omitting many details,¹³ the computational bottleneck in implementing these reducibility properties (say, instance-checkability) is indeed *polynomial interpolation* on a single variable. That is, for example, for some fixed $x_2, \dots, x_{m(n)} \in \mathbb{F}_n$, we wish to interpolate a polynomial $p: \mathbb{F}_n \rightarrow \mathbb{F}_n$ such that $p(x) = P_n(x, x_2, \dots, x_{m(n)})$ for every $x \in \mathbb{F}_n$, given oracle access to P_n .

A direct algorithm for the task above first queries P_n to obtain $P_n(z_\ell, x_2, \dots, x_{m(n)})$ for $\ell \in [d+1]$, where z_ℓ is the ℓ -th element in \mathbb{F}_n and d is the degree of P_n , and then uses Lagrange interpolation to obtain the description of p . Unfortunately, the Lagrange interpolation (as well as other interpolation methods) requires multiplying at least d elements from \mathbb{F}_n together. Since $d \geq n$ in [6], we will need a TC^0 circuit [17, 16] to compute the interpolated polynomial p .

To improve the complexity of computing p to $\text{AC}^0[2]$, we observe that for the argument above to work, d only has to be greater than the *maximum individual degree* of P_n as opposed to the total degree.¹⁴ Moreover, interpolation over \mathbb{F}_n for a constant-degree polynomial can be done in $\text{AC}^0[2]$. Hence, we made several careful non-trivial modifications to the language L^{Che19} so that the corresponding polynomials always have a constant maximum individual degree while preserving the required $\text{AC}^0[2]$ downward self-reducibility. These modifications allow us to implement the instance checker in $\text{AC}^0[2]$; see the full version for more detail.

Improved derandomization of MA_{ACC^0} . Next we discuss the intuition behind the proof of Theorem 2.

Fix $d_*, m_* \in \mathbb{N}_{\geq 1}$, our goal is to derandomize $\text{MA}_{\text{AC}_{d_*}^0[m_*]}$ into $\text{NTIMEGUESS}[2^{\text{poly} \log(n)}, \text{poly}(n)]$. Let $\mathcal{C} = \text{AC}_{d_*}^0[m_*]$. For the sake of gaining intuition, let us assume that we have a *magical PRG* construction G , such that when given a function $f: \{0, 1\}^n \rightarrow \{0, 1\}$ that is *worst-case* hard against $S(n)$ -size \mathcal{C} circuits, G^f takes $O(n)$ bits of seeds and fools all $S(n)$ -size \mathcal{C} circuits.¹⁵ Then we can start from the worst-case witness lower bound against \mathcal{C} from [33]. Roughly speaking, [33] proved that there exists $\varepsilon \in (0, 1)$ and a linear time algorithm $V_{\text{tt}}: \{0, 1\}^* \rightarrow \{0, 1\}$, such that for infinitely many $n \in \mathbb{N}$, $V_{\text{tt}}(\text{tt}(f)) = 1$ for some $f: \{0, 1\}^n \rightarrow \{0, 1\}$,¹⁶ and for every $f: \{0, 1\}^n \rightarrow \{0, 1\}$ such that $V_{\text{tt}}(\text{tt}(f)) = 1$, we know that f has no 2^{n^ε} -size \mathcal{C} circuits. Again, for the sake of intuition, we assume that the condition above holds for all $n \in \mathbb{N}$, instead of only for infinitely many $n \in \mathbb{N}$.

Making these two unrealistic assumptions, we can easily derandomize $\text{MATIME}_{\mathcal{C}}[n]$. Let $L \in \text{MATIME}_{\mathcal{C}}[n]$ and $V(x, y, z)$ be its verifier, such that $x \in L$ implies that there exists $y \in \{0, 1\}^n$ such that $\Pr_z[V(x, y, z) = 1] \geq 2/3$, and $x \notin L$ implies that for all $y \in \{0, 1\}^n$ we have $\Pr_z[V(x, y, z) = 1] \leq 1/3$. We can construct the following new verifier $V'(x, (y, f))$,

¹³These polynomials are derived from the proof of $\text{IP} = \text{PSPACE}$ [21, 29], following [31].

¹⁴The individual degree of a polynomial p with respect to a variable x_i , is the largest power of x_i appearing in a monomial of p .

¹⁵This PRG is too-good-to-be-true for two reasons: it starts from worst-case hardness instead of average-case hardness, and the circuit size it fools has no loss compared to its hardness. But we only use it to highlight the key intuition behind our proof.

¹⁶For a function $f: \{0, 1\}^n \rightarrow \{0, 1\}$, $\text{tt}(f)$ denotes the 2^n -length string that represents the truth-table of f .

where $2^{\log^\varepsilon |f|} = n$ (i.e., $|f| = 2^{\log^{1/\varepsilon} n}$.) V' first verifies that $V_{\text{tt}}(f) = 1$, and then verifies that $\Pr_{s \in O(\log |f|)}[V(x, y, G^f(s)) = 1] \geq 1/2$. We can see that V' runs in quasi-polynomial time, and indeed V' puts $L \in \text{NQP}$.

However, the derandomization above requires that the whole truth-table f is given as the witness, which has length $|f| = 2^{\text{poly}(\log n)}$. To improve this, we consider a thought experiment: letting $\ell = \log |f| = \log^{1/\varepsilon} n$, what if for *some* $f: \{0, 1\}^\ell \rightarrow \{0, 1\}$ such that $V_{\text{tt}}(\text{tt}(f)) = 1$, f indeed has a $2^{2 \cdot \ell^\varepsilon}$ -size \mathcal{C} circuit? Assuming this is true, then we observe that we do not have to guess the whole truth-table f in our verifier V' anymore, and can instead just guess a $2^{2 \cdot \log^\varepsilon |f|} = \text{poly}(n)$ -size circuit $C: \{0, 1\}^{\log^{1/\varepsilon} n} \rightarrow \{0, 1\}$ and use its truth-table as f ! Hence, in this thought experiment, we dramatically reduce our witness length from $2^{\text{poly}(\log n)}$ to $\text{poly}(n)$.

Of course, what if the hypothesis in our thought experiment is not true? Namely, what if for *every* $f: \{0, 1\}^\ell \rightarrow \{0, 1\}$ such that $V(\text{tt}(f)) = 1$, f has no $2^{2 \cdot \ell^\varepsilon}$ -size \mathcal{C} circuits as well? Staring at this for a moment, one can realize that now V certifies not only 2^{ℓ^ε} hardness, but indeed $2^{2 \cdot \ell^\varepsilon}$ hardness! This essentially means that we can keep doing this argument and eventually obtain a quasi-polynomial-time derandomization of $\text{MA}_{\mathcal{C}}$ with only $\text{poly}(n)$ witnesses.

Of course, the above is just an idealized setting that captures our key ideas; see the full version for detailed proofs of how we managed to get rid of the “magic PRG assumption” using machinery developed in [7].

2 Overview of the Derandomization-centric Perspective on the Algorithmic Method

This section aims to give a detailed overview of the derandomization-centric perspective on the algorithmic method.

In Subsection 2.1, we will first provide an overview of the original proofs from [39] and [23]. We will give a somewhat different presentation from that of [39, 23]. Our presentation is centered around the concept of *easy witness lemmas (EWLs)*, which converts *witness lower bounds* into *circuit lower bounds for nondeterministic time classes*. Thus, we can decompose the proof into two parts: first, we prove a witness lower bound; second, we apply an easy witness lemma to convert the obtained witness lower bound into a circuit lower bound for nondeterministic time classes.

Next, in Subsection 2.2, we give an overview of our results on circuit lower bounds for nondeterministic time classes, which follows a *derandomization-centric perspective*. Roughly speaking, our proofs are centered around *derandomization of Merlin-Arthur classes*. Our proofs for lower bounds for nondeterministic time classes can also be naturally decomposed into two parts: first, we prove a circuit lower bound for a certain subclass of Merlin-Arthur protocols; second, we derandomize the same subclass into a nondeterministic time class. To obtain average-case circuit lower bounds for nondeterministic time classes, it amounts to start from average-case lower bounds for Merlin-Arthur classes and apply a careful win-win analysis (adopted from [9]).

We first recall the definitions of the following standard derandomization problems.

- **CAPP (Circuit Acceptance Probability Problem) with error δ (denoted CAPP_δ):** Given a circuit C on n inputs, estimate $\Pr_{x \in_{\text{R}} \{0, 1\}^n}[C(x) = 1]$ within an *additive error* of δ . When not explicitly stated, δ is set to be $1/3$ by default.
- **CAPP with inverse-circuit-size error (denoted as $\widetilde{\text{CAPP}}$):** Given a circuit C of size S on n input bits, estimate $\Pr_{x \in \{0, 1\}^n}[C(x) = 1]$ within an additive error of $1/S$.

Notation. We use \mathbb{N} to denote all non-negative integers and $\mathbb{N}_{\geq 1}$ to denote all positive integers. We say a circuit class \mathcal{C} is **concrete** if we can talk about the \mathcal{C} complexity of a single function $f: \{0, 1\}^n \rightarrow \{0, 1\}$ (as opposed to a family of functions $\{f_n\}_{n \in \mathbb{N}_{\geq 1}}$). For example, for fixed $d, m \in \mathbb{N}_{\geq 1}$, $\text{AC}_d^0[m]$ is a concrete circuit class, but AC^0 is not (because the depth can vary). We say a concrete circuit class \mathcal{C} is *typical* if it is closed under (1) taking the negation of the output, (2) taking the projections of the input, and (3) flipping input bits.

2.1 An Overview of Williams' EWL-centered Proofs

We begin by formally stating the generic connection between non-trivial circuit analysis algorithms and lower bounds from [37, 39, 23].¹⁷

► **Theorem 8** ([37, 39]). *Let \mathcal{C} be a typical concrete circuit class. If there is a $2^n/n^{\omega(1)}$ -time algorithm for CAPP of poly(n)-size n -input $\text{AC}_2^0 \circ \mathcal{C}$ circuits, then $\text{NE} \not\subseteq \mathcal{C}$.*

► **Theorem 9** ([23]). *Let \mathcal{C} be a typical concrete circuit class and $\varepsilon \in (0, 1)$. If CAPP for 2^{n^ε} -size $\text{AC}_2^0 \circ \mathcal{C}$ can be solved in $2^n/n^{\omega(1)}$ time, then $\text{NQP} \not\subseteq \mathcal{C}$.*

Notation. Let $s: \mathbb{N} \rightarrow \mathbb{N}$ and \mathcal{C} be a concrete circuit class. We say that **NE** *does not admit $s(n)$ -size \mathcal{C} witnesses*, if there exists a verifier $V(x, y)$ that takes input $|x| = n$, $|y| = 2^n$ and runs in $2^{O(n)}$ time, such that for infinitely many $x \in \{0, 1\}^*$, the following hold:

1. there exists $y \in \{0, 1\}^{2^{|x|}}$ such that $V(x, y) = 1$;
2. for every $y \in \{0, 1\}^{2^{|x|}}$ such that $V(x, y) = 1$, it follows that $\text{func}(y)$ has no $s(n)$ -size \mathcal{C} circuit.¹⁸

Moreover, we say that *unary NE* *does not admit $s(n)$ -size \mathcal{C} witnesses*, if for some verifier V and for infinitely many $n \in \mathbb{N}_{\geq 1}$, the above two conditions hold for $x = 1^n$. This is a stronger statement and implies that **NE** does not admit $s(n)$ -size \mathcal{C} witnesses.

2.1.1 Overview for the proof of Theorem 8

The easy witness lemma of [18] says the following:

► **Lemma 10** (EWL for **NE** [18]). *Let \mathcal{C} be a typical concrete circuit class. If **NE** does not admit poly(n)-size \mathcal{C} witnesses¹⁹, then $\text{NE} \not\subseteq \mathcal{C}$.*

Indeed, the original statement says the contrapositive of Lemma 10: if $\text{NE} \subseteq \mathcal{C}$, then **NE** admits polynomial-size \mathcal{C} witnesses. Hence the name of easy witness lemma (*i.e.*, **NE** admits small circuit implies that **NE** admits *easy witnesses*).²⁰ We present it in this way since it is clear that witness lower bounds imply circuit lower bounds.

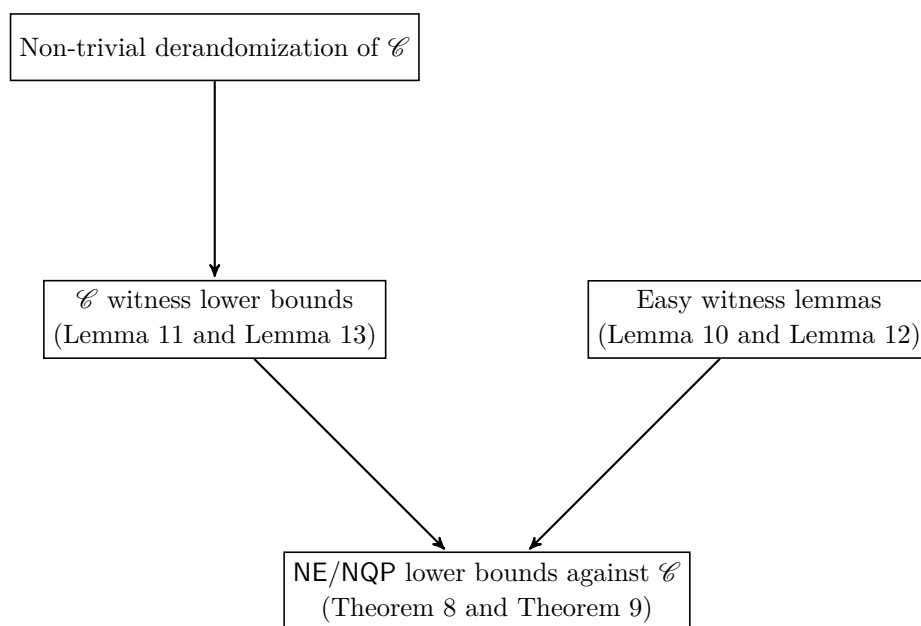
Williams then gave a way to prove witness lower bounds from non-trivial derandomization. The lemma below is from [38], but the proof ideas are similar to that from [37, 39].

¹⁷We remark that in Theorem 8 and Theorem 9, CAPP can be replaced by **Gap-UNSAT**: a circuit-analysis problem that is weaker than both CAPP and SAT (see [23] for details). We will work with CAPP for simplicity.

¹⁸Here we use $\text{func}(y)$ to denote the $|x|$ -bit Boolean function whose truth-table is y .

¹⁹More precisely, **NE** does not admit n^k -size \mathcal{C} witnesses for every $k \in \mathbb{N}_{\geq 1}$.

²⁰[18] indeed talks about **NEXP** instead of **NE**; we choose to work with **NE** since it simplifies the discussions.



■ **Figure 1** High-level structure of Williams' EWL-centered proofs.

► **Lemma 11** ([38]). *Let \mathcal{C} be a typical concrete circuit class. If CAPP for polynomial-size $\text{AC}_2^0 \circ \mathcal{C}$ circuits can be solved in $2^n/n^{\omega(1)}$ time, then unary NE does not admit $\text{poly}(n)$ -size \mathcal{C} witnesses.*

Combining Lemma 10 and Lemma 11 immediately proves Theorem 8.

2.1.2 Overview for the Proof of Theorem 9

To obtain lower bounds for NQP, [23] proved the following easy witness lemma. Again, we state their lemma in the contrapositive.

► **Lemma 12** (EWL for NQP [23]). *Let \mathcal{C} be a typical concrete circuit class. If NE does not admit 2^{n^ε} -size \mathcal{C} witnesses for some $\varepsilon \in (0, 1)$, then $\text{NQP} \not\subseteq \mathcal{C}$.*

We note that the lemma above is weaker than the easy witness lemma for NQP in [23],²¹ but we observe that it still suffices for circuit lower bounds for NQP. To obtain $\text{NQP} \not\subseteq \mathcal{C}$, we need the following adaption of Lemma 11.

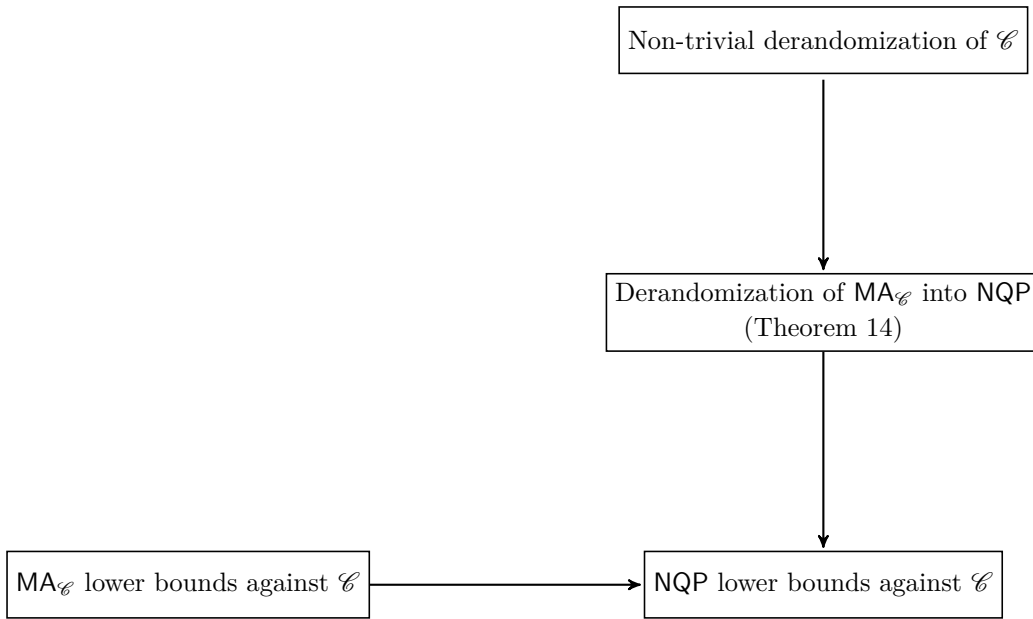
► **Lemma 13** ([38]). *Let \mathcal{C} be a typical concrete circuit class and $\varepsilon \in (0, 1)$. If CAPP for 2^{n^ε} -size $\text{AC}_2^0 \circ \mathcal{C}$ can be solved in $2^n/n^{\omega(1)}$ time, then unary NE does not admit $2^{n^\varepsilon/2}$ -size \mathcal{C} witnesses.*

Now, combining Lemma 12 and Lemma 13, Theorem 9 follows immediately.

2.2 NQP Lower Bounds via Derandomization of Merlin-Arthur Protocols

In the rest of this section, we give outlines of our alternative proofs of the following two results: (1) $\text{NQP} \not\subseteq \text{ACC}^0$ [23] and (2) there is a constant $\beta \in \mathbb{N}_{\geq 1}$ such that $\text{NTIME}[2^{\log^\beta n}]$

²¹Its contrapositive says that if $\text{NQP} \subset \mathcal{C}$, then NE admits 2^{n^ε} -size \mathcal{C} witnesses for every $\varepsilon \in (0, 1)$; this consequence is weaker than NQP admits polynomial-size \mathcal{C} witnesses.



■ **Figure 2** High-level structure of the new derandomization-centric perspective for proving NTIME lower bounds.

cannot be $1/2 + 1/\text{poly}(n)$ -approximated by $\text{poly}(n)$ -size ACC^0 circuits [9].²² See Theorem 19 and Theorem 23 for detailed statements. In the full version, we prove stronger versions of Theorem 19 and Theorem 23 using our improved NPRG construction from, but the proof outlines are identical.

Derandomization of $\text{MA}_{\mathcal{C}}$. Recall that $\text{MA}_{\mathcal{C}}$ denotes the sub-class of Merlin-Arthur protocols whose verification can be simulated by \mathcal{C} circuits for every possible witness, and NPRG is a weaker version of PRG that suffices to derandomize Merlin-Arthur protocols; see the full version for formal definitions.

The following theorem is from [7].

► **Theorem 14** ([7, Theorem 7.1]). *Let \mathcal{C} be a typical concrete circuit class and $\varepsilon \in (0, 1)$. Suppose that CAPP of 2^{n^ε} -size $\text{AND}_4 \circ \mathcal{C} \circ \text{AC}_2^0$ circuits can be solved in 2^{n-n^ε} time. Then there is a $\delta \in (0, 1)$ and an infinity often NPRG for 2^{n^δ} -size \mathcal{C} circuits with error 2^{-n^δ} , seed-length $\text{poly}(n)$, and $2^{\text{poly}(n)}$ running time. Consequently, $\text{MA}_{\mathcal{C}} \subseteq \text{i.o. NTIME}[2^{\log^\beta n}]$ for some $\beta \in \mathbb{N}_{\geq 1}$.*

2.2.1 NQP Lower Bounds via Derandomization

In order to apply Theorem 14 to get circuit lower bounds for NQP, (roughly speaking) we will prove an $\text{MA}_{\mathcal{C}}$ lower bounds against \mathcal{C} . In more detail, we will prove the following theorem.

²²We note that both of [23] and [9] indeed proved NQP lower bounds against $2^{\log^a n}$ -size ACC^0 circuits for every $a \in \mathbb{N}_{\geq 1}$. We only consider lower bounds against all polynomial-size ACC^0 circuits in this paper for simplicity, but our proofs can be straightforwardly modified to prove the same lower bounds from [23, 9].

► **Theorem 15.** *Let \mathcal{C} be a typical concrete circuit class. There is a universal constant $d_v \in \mathbb{N}_{\geq 1}$ such that for all $a \in \mathbb{N}_{\geq 1}$, it holds that*

$$\left(\text{MA}_{\text{AC}_{d_v}^0[2] \circ \mathcal{C}}\right)_{/1} \not\subseteq \mathcal{C}\text{-SIZE}[n^a].$$

It seems that assuming the required $\widetilde{\text{CAPP}}$ algorithm for $\text{AC}_{d_v+1}^0[2] \circ \mathcal{C} \circ \text{AC}_2^0$ and applying Theorem 14,²³ we will be able to derandomize the hard $\left(\text{MA}_{\text{AC}_{d_v}^0[2] \circ \mathcal{C}}\right)_{/1}$ language from Theorem 15 into $\text{NTIME}[2^{\log^\beta n}]$ (we ignore the extra one bit of advice in the hard language for now), which *should* imply $\text{NQP} \not\subseteq \mathcal{C}$. However, there is a huge caveat that we explain below.

Retaining the hardness after derandomization. The issue is that the $\text{MA}_{/1}$ hard language L from Theorem 15 is only *infinitely often* hard, meaning that we only know for infinitely many input lengths $n \in \mathbb{N}_{\geq 1}$, L_n is hard against \mathcal{C} circuits. Also, the derandomization of Theorem 14 works *infinitely often* too, in the sense that our new NQP language L' only agrees with the hard language L on infinitely many input lengths n . Let I_{hard} and I_{derand} be the input lengths that L_n is hard and $L'_n = L_n$, respectively. We see that it is possible that $I_{\text{hard}} \cap I_{\text{derand}} = \emptyset$, meaning that our new NQP language L' *does not retain any hardness of L* .

Following [23], the idea is to make both I_{hard} and I_{derand} larger so that they *must intersect at infinitely many input lengths*.

In more detail, we first strengthen Theorem 14 by showing the following theorem.

► **Theorem 16.** *Let \mathcal{C} be a typical concrete circuit class. Suppose that there is a constant $\varepsilon \in (0, 1)$ and an infinity often NPRG for 2^{n^ε} -size \mathcal{C} circuits with $\text{poly}(n)$ seed-length and $2^{\text{poly}(n)}$ running time.*

Then, there is a constant $\beta \in \mathbb{N}_{\geq 1}$ that only depends on ε such that for every $L \in (\text{MA}_{\mathcal{C}})_{/1}$ and $c \in \mathbb{N}_{\geq 1}$, there is an $L' \in \text{NTIME}[2^{\log^\beta n}]_{/O(\log \log n)}$ such that for infinitely many $n \in \mathbb{N}$, for every $m \in [n, n^c]$, L and L' agree on all m -bit inputs.

Combining Theorem 14 and Theorem 16, we immediately have the following strengthening of Theorem 14.

► **Corollary 17.** *Let \mathcal{C} be a typical concrete circuit class and $\varepsilon \in (0, 1)$. Assuming the hypothesized $\widetilde{\text{CAPP}}$ algorithm from Theorem 14, the conclusion of Theorem 16 holds.*

Roughly speaking, Corollary 17 says that by allowing $O(\log \log n)$ bits of advice, we can enlarge I_{derand} from a set of infinitely many integers to a union of infinitely many segments of the form $[n, n^c]$, where c is a constant of our choice.

Next, we have the following strengthening of Theorem 15, which fits perfectly with the larger I_{derand} above.

► **Theorem 18.** *Let \mathcal{C} be a typical concrete circuit class. There is a universal constant $d_v \in \mathbb{N}_{\geq 1}$ such that for all $a \in \mathbb{N}_{\geq 1}$, there is a constant $c \in \mathbb{N}_{\geq 1}$ and a language $L \in \left(\text{MA}_{\text{AC}_{d_v}^0[2] \circ \mathcal{C}}\right)_{/1}$ such that, for all large enough $n \in \mathbb{N}_{\geq 1}$, there exists $m \in [n, n^c]$ such that L_m does not have m^a -size \mathcal{C} circuits.*

²³We recommend reader to think of $\mathcal{C} = \text{AC}_d^0[6]$ for some large constant $d \in \mathbb{N}_{\geq 1}$, then we only need non-trivial $\widetilde{\text{CAPP}}$ for $\text{AC}_{d+O(1)}^0[6]$, which follows from [39, 34].

Essentially, it says that we can enlarge I_{hard} to be a *hitting set* for all segment $[n, n^c]$: for every large enough $n \in \mathbb{N}_{\geq 1}$, $[n, n^c] \cap I_{\text{hard}} \neq \emptyset$. This fits perfectly with the I_{derand} above, which consists of infinitely many segments of the form $[n, n^c]$. Hence, we have that $I_{\text{hard}} \cap I_{\text{derand}}$ is an infinite set.

Therefore, combining Corollary 17 and Theorem 18, we immediately have the following theorem.²⁴

► **Theorem 19.** *Let \mathcal{C} be a typical concrete circuit class, $\varepsilon \in (0, 1)$, and $d_v \in \mathbb{N}_{\geq 1}$ be the constant from Theorem 18. Suppose that $\widetilde{\text{CAPP}}$ of 2^{n^ε} -size $\text{AC}_{d_v+1}^0[2] \circ \mathcal{C} \circ \text{AC}_2^0$ circuits can be solved in 2^{n-n^ε} -time. Then $\text{NQP} \not\subseteq \mathcal{C}$.*

As a corollary from the theorem above and Williams' #SAT algorithm for ACC^0 [39, 34], we immediately have that $\text{NQP} \not\subseteq \text{AC}_{d_*}^0[m_*]$ for every $d_*, m_* \in \mathbb{N}$, which implies $\text{NQP} \not\subseteq$, as discussed in Subsubsection 1.1.1.

2.2.2 Average-case Lower Bounds for NQP via Randomized Encodings

Finally, we strengthen Theorem 19 to give average-case lower bounds as well. Given the discussions above, it seems that we can simply strengthen the $\text{MA}_{\mathcal{C}}$ lower bounds of Theorem 18 to average-case, and our derandomization from Corollary 17 would immediately imply average-case lower bounds for NQP.

We are indeed able to strengthen Theorem 18 to an average-case lower bound, but only with very weak inapproximability.

► **Theorem 20.** *Let \mathcal{C} be a typical concrete circuit class. There are universal constants $d_v, \tau \in \mathbb{N}_{\geq 1}$ such that for all $a \in \mathbb{N}_{\geq 1}$, there is a constant $c \in \mathbb{N}_{\geq 1}$ and a language $L \in \left(\text{MA}_{\text{AC}_{d_v}^0[2] \circ \mathcal{C}} \right)_{/1}$ such that, for all large enough $n \in \mathbb{N}_{\geq 1}$, there exists $m \in [n, n^c]$ such that L_m cannot be $(1 - m^{-\tau})$ -approximated by m^a -size \mathcal{C} circuits.²⁵*

Combining with Corollary 17, we immediately have the following theorem.

► **Theorem 21.** *Let \mathcal{C} be a typical concrete circuit class, $\varepsilon \in (0, 1)$, and $d_v, \tau \in \mathbb{N}_{\geq 1}$ be the constants from Theorem 20. Suppose that $\widetilde{\text{CAPP}}$ of 2^{n^ε} -size $\text{AC}_{d_v+1}^0[2] \circ \mathcal{C} \circ \text{AC}_2^0$ circuits can be solved in 2^{n-n^ε} time. Then NQP cannot be $(1 - n^{-\tau})$ -approximated by $\text{poly}(n)$ -size \mathcal{C} circuits.*

To improve the inapproximability of Theorem 21 from $1 - n^{-\tau}$ to $1/2 + 1/\text{poly}(n)$, we wish to perform some *mild-to-strong average-case hardness amplification* (e.g., an XOR Lemma). Unfortunately, we currently *do not* have such an amplification for weak circuit classes such as ACC^0 (and there are barriers against such possibilities, see, e.g., [28, 13]).

Chen and Ren [9] overcame the issue above with a clever win-win argument based on *randomized encodings* [19, 2] and *approximate linear sums*. Recall that for a $\text{Sum} \circ \mathcal{C}$ circuit $L = \sum_{i \in [m]} \alpha_i \cdot C_i$ (where each C_i is a \mathcal{C} circuit), the complexity of L is defined as

$$\max \left(\sum_{i \in [m]} |\alpha_i|, \sum_{i \in [m]} \text{SIZE}(C_i) \right).$$

²⁴A direct application of Corollary 17 yields a hard language in $\text{NQP}_{/O(\log \log n)}$ instead of just NQP. Those advice can nonetheless be removed via a straightforward *enumeration trick* (from [11]).

²⁵We remark that in the full version we will indeed prove a stronger version where the hard language L is in $\left((\text{MA} \cap \text{coMA})_{\text{AC}_{d_v}^0[2] \circ \mathcal{C}} \right)_{/1}$. We will discuss why this is needed at the end of this subsection.

For a function $F: \{0, 1\}^n \rightarrow \{0, 1\}$, we say that F admits a $\widetilde{\text{Sum}}_\delta \circ \mathcal{C}$ circuit of complexity S , if there exists a $\text{Sum} \circ \mathcal{C}$ circuit L with complexity at most S , such that $|F(x) - L(x)| \leq \delta$ for every $x \in \{0, 1\}^n$.

Using the techniques from randomized encodings, [9] proved the following win-win result.

► **Lemma 22** ([9]). *Let \mathcal{C} be a typical concrete circuit class. There is a language $L \in \text{P}$ such that one of the following holds:*

1. *For every $k \in \mathbb{N}_{\geq 1}$, L cannot be $(1/2 + n^{-k})$ -approximated by n^k -size \mathcal{C} circuits.*
2. *There is a constant $\gamma \in \mathbb{N}_{\geq 1}$ such that every S -size formula admits a $\widetilde{\text{Sum}}_{0.01} \circ \mathcal{C}$ circuit of complexity S^γ .*

In other words, either (Item (1)) we already have strongly average-case lower bounds for P against \mathcal{C} , or (Item (2)) formulas can be simulated by $\widetilde{\text{Sum}} \circ \mathcal{C}$ with a polynomial blow-up in size. The key observation now is that an NPRG that fools \mathcal{C} circuits with a small error also fools functions admitting low-complexity $\widetilde{\text{Sum}} \circ \mathcal{C}$ circuits. Hence, now we are able to perform the following win-win analysis:

- Suppose Item (1) of Lemma 22 holds. Then it immediately follows that NQP cannot be $(1/2 + 1/\text{poly}(n))$ -approximated by $\text{poly}(n)$ -size \mathcal{C} circuits.
- Otherwise, Item (2) of Lemma 22 holds. Then under the condition of Theorem 14, we would have i.o. NPRG for formulas (note that the i.o. NPRG from Theorem 14 indeed has a small error). Applying Theorem 16, this implies that we can now derandomize $\text{MA}_{\text{Formula}}$ as follows:

There is a constant $\beta \in \mathbb{N}_{\geq 1}$ such that for every $L \in (\text{MA}_{\text{Formula}})_{/1}$ and every $c \in \mathbb{N}_{\geq 1}$, there is an $L' \in \text{NTIME}[2^{\log^\beta n}]_{/O(\log \log n)}$ such that for infinitely many $n \in \mathbb{N}$, for every $m \in [n, n^c]$, L and L' agree on all m -bit inputs.

Noting that formulas are closed under taking an $\text{AC}^0[2]$ circuit at the top, we now can use the derandomization above together with Theorem 20 to obtain an NQP language that is $(1 - n^{-\tau})$ -hard against polynomial-size formulas, which can then be amplified to $(1/2 + 1/\text{poly}(n))$ -hardness against formulas, using mild-to-average-case hardness amplification for formulas.

If we further assume that \mathcal{C} can be simulated by Formula , then we also have that NQP cannot be $1/2 + 1/\text{poly}(n)$ -approximated by $\text{poly}(n)$ -size \mathcal{C} circuits.

To summarize, we have the following theorem.

► **Theorem 23** (strong average-case lower bound for NQP via an additional win-win argument). *Let \mathcal{C} be a typical concrete circuit class that can be simulated by formulas. Suppose that for some $\eta \in (0, 1)$, $\widetilde{\text{CAPP}}$ of 2^{n^η} -size $\text{AND}_4 \circ \mathcal{C} \circ \text{AC}_2^0$ circuits can be deterministically solved in 2^{n-n^η} time. Then, there is $\beta \in \mathbb{N}_{\geq 1}$ such that $\text{NTIME}[2^{\log^\beta n}]$ cannot be $1/2 + 1/\text{poly}(n)$ -approximated by $\text{poly}(n)$ -size \mathcal{C} circuits.*

Note that applying the theorem above directly only shows that for every $d_*, m_* \in \mathbb{N}_{\geq 1}$, there is $\beta \in \mathbb{N}_{\geq 1}$ that depends on d_*, m_* such that $\text{NTIME}[2^{\log^\beta n}]$ cannot be $1/2 + 1/\text{poly}(n)$ -approximated by $\text{poly}(n)$ -size $\text{AC}_{d_*}^0[m_*]$ circuits. To swap the quantifiers before d_*, m_* and β , we can apply another win-win analysis from [9]; see the full version for details.

Finally, we give one technical remark.

Mild-to-strong hardness amplification requires $(\text{N} \cap \text{coN})\text{QP}$ lower bounds. Recall that we wish to apply an XOR Lemma to the mildly average-case hard NQP language L from Theorem 21. This causes a subtle issue: $L^{\oplus 2}(x, y) := L(x) \oplus L(y)$ may not be in NQP, since to certify $L^{\oplus 2}(x, y) = 1$, one needs to prove exactly one of $L(x)$ and $L(y)$ is 1 and the other one is 0; we cannot prove (say) $L(y) = 0$ since this requires that $L \in \text{coNQP}$.

To resolve this issue, we wish to get a mildly average-case hard language L from $(\mathsf{N} \cap \mathsf{coN})\mathsf{QP}$ instead of NQP . It is easy to see that the derandomization from Corollary 17 also derandomize $(\mathsf{MA} \cap \mathsf{coMA})_{/1}$ languages into $(\mathsf{N} \cap \mathsf{coN})\mathsf{QP}_{/O(\log \log n)}$ languages. Hence, we strengthen Theorem 20 so that the hard languages now belong to $(\mathsf{MA} \cap \mathsf{coMA})_{/1}$.

References

- 1 M. Ajtai. Σ_1^1 -formulae on finite structures. *Annals of Pure and Applied Logic*, 24(1):1–48, 1983.
- 2 Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography in NC^0 . *SIAM J. Comput.*, 36(4):845–888, 2006. doi:10.1137/S0097539705446950.
- 3 Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998. doi:10.1145/278298.278306.
- 4 Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: A new characterization of NP . *J. ACM*, 45(1):70–122, 1998. doi:10.1145/273865.273901.
- 5 Eli Ben-Sasson and Emanuele Viola. Short PCPs with projection queries. In *Proc. 41st International Colloquium on Automata, Languages and Programming (ICALP)*, pages 163–173, 2014.
- 6 Lijie Chen. Non-deterministic quasi-polynomial time is average-case hard for ACC circuits. In *Proc. 60th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 1281–1304, 2019.
- 7 Lijie Chen, Xin Lyu, and Richard Ryan Williams. Almost-everywhere circuit lower bounds from non-trivial derandomization. In *Proc. 61st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 1–12, 2020.
- 8 Lijie Chen and Hanlin Ren. Strong average-case lower bounds from non-trivial derandomization. In *Proc. 52nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 1327–1334, 2020.
- 9 Lijie Chen and Hanlin Ren. Strong average-case circuit lower bounds from nontrivial derandomization. *SIAM Journal of Computing*, 51(3):STOC20–115, 2021.
- 10 Lijie Chen and R. Ryan Williams. Stronger Connections Between Circuit Analysis and Circuit Lower Bounds, via PCPs of Proximity. In *Proc. 34th Annual IEEE Conference on Computational Complexity (CCC)*, pages 19:1–19:43, 2019.
- 11 Ruiwen Chen, Igor Carboni Oliveira, and Rahul Santhanam. An average-case lower bound against ACC^0 . In *LATIN 2018: Theoretical Informatics - 13th Latin American Symposium, Proceedings*, pages 317–330, 2018.
- 12 Merrick Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984.
- 13 Aryeh Grinberg, Ronen Shaltiel, and Emanuele Viola. Indistinguishability by adaptive procedures with advice, and lower bounds on hardness amplification proofs. In *Proc. 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 956–966, 2018.
- 14 Joshua A. Grochow and Toniann Pitassi. Circuit complexity, proof complexity, and polynomial identity testing: The ideal proof system. *J. ACM*, 65(6):37:1–37:59, 2018. doi:10.1145/3230742.
- 15 Johan Håstad. Almost optimal lower bounds for small depth circuits. *Advances in Computing Research*, 5:143–170, 1989.
- 16 Alexander Healy and Emanuele Viola. Constant-depth circuits for arithmetic in finite fields of characteristic two. In *Proc. 23rd Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 672–683, 2006.
- 17 William Hesse, Eric Allender, and David A. Mix Barrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *J. Comput. Syst. Sci.*, 65(4):695–716, 2002.

- 18 Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: exponential time vs. probabilistic polynomial time. *Journal of Computer and System Sciences*, 65(4):672–694, 2002.
- 19 Yuval Ishai and Eyal Kushilevitz. Perfect constant-round secure computation via perfect randomizing polynomials. In *Proc. 29th International Colloquium on Automata, Languages and Programming (ICALP)*, pages 244–256, 2002. doi:10.1007/3-540-45465-9_22.
- 20 Hamid Jahanjou, Eric Miles, and Emanuele Viola. Local reductions. In *Proc. 42nd International Colloquium on Automata, Languages and Programming (ICALP)*, pages 749–760, 2015.
- 21 Carsten Lund, Lance Fortnow, Howard Karloff, and Noam Nisan. Algebraic methods for interactive proof systems. *Journal of the Association for Computing Machinery*, 39(4):859–868, 1992.
- 22 Cody Murray and R. Ryan Williams. Circuit lower bounds for nondeterministic quasi-polytime: An easy witness lemma for NP and NQP. In *Proc. 50th Annual ACM Symposium on Theory of Computing (STOC)*, pages 890–901, 2018.
- 23 Cody D. Murray and R. Ryan Williams. Circuit lower bounds for nondeterministic quasi-polytime from a new easy witness lemma. *SIAM Journal of Computing*, 49(5), 2020.
- 24 Noam Nisan and Avi Wigderson. Hardness vs. randomness. *Journal of Computer and System Sciences*, 49(2):149–167, 1994.
- 25 Alexander A. Razborov. Lower bounds on the size of constant-depth networks over a complete basis with logical addition. *Mathematical Notes of the Academy of Science of the USSR*, 41(4):333–338, 1987.
- 26 Rahul Santhanam. Circuit lower bounds for Merlin-Arthur classes. *SIAM Journal of Computing*, 39(3):1038–1061, 2009.
- 27 Joel I. Seiferas, Michael J. Fischer, and Albert R. Meyer. Separating nondeterministic time complexity classes. *J. ACM*, 25(1):146–167, 1978. doi:10.1145/322047.322061.
- 28 Ronen Shaltiel and Emanuele Viola. Hardness amplification proofs require majority. *SIAM Journal of Computing*, 39(7):3122–3154, 2010.
- 29 Adi Shamir. $IP = PSPACE$. *Journal of the ACM*, 39(4):869–877, 1992.
- 30 Roman Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proc. 19th Annual ACM Symposium on Theory of Computing (STOC)*, pages 77–82, 1987.
- 31 Luca Trevisan and Salil P. Vadhan. Pseudorandomness and average-case complexity via uniform reductions. *Computational Complexity*, 16(4):331–364, 2007.
- 32 Nikhil Vyas. Unpublished manuscript, 2019.
- 33 R. Ryan Williams. Natural proofs versus derandomization. *SIAM Journal of Computing*, 45(2):497–529, 2016.
- 34 Richard Ryan Williams. New algorithms and lower bounds for circuits with linear threshold gates. *Theory of Computing*, 14:Paper No. 17, 25, 2018.
- 35 Ryan Williams. Improving exhaustive search implies superpolynomial lower bounds. In *Proc. 42nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 231–240, 2010.
- 36 Ryan Williams. Non-uniform ACC circuit lower bounds. In *Proc. 26th Annual IEEE Conference on Computational Complexity (CCC)*, pages 115–125, 2011.
- 37 Ryan Williams. Improving exhaustive search implies superpolynomial lower bounds. *SIAM Journal of Computing*, 42(3):1218–1244, 2013.
- 38 Ryan Williams. Natural proofs versus derandomization. In *Proc. 45th Annual ACM Symposium on Theory of Computing (STOC)*, pages 21–30, 2013.
- 39 Ryan Williams. Nonuniform ACC circuit lower bounds. *Journal of the ACM*, 61(1):2:1–2:32, 2014.
- 40 Andrew C-C. Yao. Separating the polynomial-time hierarchy by oracles. In *Proc. 26th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 1–10, 1985.
- 41 Stanislav Žák. A Turing machine time hierarchy. *Theoretical Computer Science*, 26(3):327–333, 1983.