Strong Average-Case Lower Bounds from Non-trivial Derandomization

Lijie Chen†
lijieche@mit.edu
CSAIL, MIT
Cambridge, MA, USA

Hanlin Ren
rhl16@mails.tsinghua.edu.cn
IIIS, Tsinghua University
Beijing, China

ABSTRACT

We prove that for all constants \(a\), NQP = NTIME[\(n^{\text{polylog}(n)}\)] cannot be \((1/2+2^{-\log^a n})\)-approximated by \(2^{\log^a n}\)-size ACC\(^0\) circuits. As a straightforward application, we obtain an infinitely often (NE \(\cap\) coNE)/1-computable pseudorandom generator for poly-size ACC\(^0\) circuits with seed length \(2^{\log^e n}\), for all \(e > 0\).

More generally, we establish a connection showing that, for a typical circuit class \(\mathcal{C}\), non-trivial nondeterministic algorithms estimating the acceptance probability of a given \(S\)-size \(\mathcal{C}\) circuit with an additive error \(1/S\) (we call it a CAPP algorithm) imply strong \((1/2+1/n^{\omega(1)})\) average-case lower bounds for nondeterministic time classes against \(\mathcal{C}\) circuits. Note that the existence of such (deterministic) algorithms is much weaker than the widely believed conjecture PromiseBPP = PromiseP.

We also apply our results to several sub-classes of TC\(^0\) circuits. First, we show that for all \(k\), NC\(^p\) cannot be \((1/2+n^{-k})\)-approximated by \(n^k\)-size Sum \(\oplus\) THR circuits (exact \(\oplus\) linear combination of threshold gates), improving the corresponding worst-case result in [Williams, CCC 2018]. Second, we establish strong average-case lower bounds and build (NE \(\cap\) coNE)/1-computable PRGs for Sum \(\oplus\) PTF circuits, for various regimes of degrees. Third, we show that non-trivial CAPP algorithms for MAJ \(\Rightarrow\) MAJ indeed already imply worst-case lower bounds for TC\(^0\) (MAJ \(\Rightarrow\) MAJ). Since exponential lower bounds for MAJ \(\Rightarrow\) MAJ are already known, this suggests TC\(^0\) lower bounds are probably within reach.

Our new results build on a line of recent works, including [Murray and Williams, STOC 2018], [Chen and Williams, CCC 2019], and [Chen, FOCS 2019]. In particular, it strengthens the corresponding \((1/2+1/polylog(n))\)-inapproximability average-case lower bounds in [Chen, FOCS 2019].

1 INTRODUCTION

1.1 Background and Motivation

A holy grail of theoretical computer science is to prove unconditional circuit lower bounds for explicit functions (such as \(NP \not\subset P/poly\)). To approach this notoriously hard central open problem, the first step is to understand the power of various constant depth circuit classes. Back in the 1980s, there was a lot of significant progress in proving lower bounds for constant depth circuits. A line of works [2, 22, 28, 53] established exponential lower bounds for \(AC^0\) (constant depth circuits consisting of AND/OR gates of unbounded fan-in), and [36, 41] proved exponential lower bounds for \(AC^0[p]\) (\(AC^0\) circuits extended with MOD\(_p\) gates) when \(p\) is a prime.

However, the progress had stopped there—the power of \(AC^0[m]\) for a composite \(m\) had been elusive, despite that it had been conjectured that they cannot even compute the majority function. In fact, it had been a notorious long-standing open question in computational complexity whether \(NEXP\) (nondeterministic exponential time) has polynomial-size \(AC^0\) circuits, until a seminal work by Williams [49] a few years ago, which proved NEXP does not have polynomial-size \(AC^0\) circuits, via a new algorithmic approach to circuit lower bounds [47].

Not only being an exciting new development after a long gap, the new circuit lower bound is also remarkable as it surpasses all previous known barriers for proving circuit lower bounds: relativization [11], algebrization [1], and natural proofs [37]. Moreover, the

KEYWORDS
circuit complexity, average-case complexity, derandomization

ACM Reference Format:
underlying method (the algorithmic method) puts many important classical complexity gems together, ranging from nondeterministic time hierarchy theorem [38, 54], IP = PSPACE [32, 40], hardness vs randomness [35], to PCP Theorem [7, 8].

Recent development of the algorithmic approach to circuit lower bounds. Recently, Murray and Williams [34] significantly advanced the algorithmic approach by proving that strong enough circuit-analysis (Gap-UNSAT)\(^\text{a}\) algorithms can also imply circuit lower bounds for NQP (nondeterministic quasi-polynomial time) or NP, instead of the previous gigantic class NEXP. Building on the new connection and the corresponding algorithms for ACC\(^0\) \(\circ\) THR [48], they showed that NQP \(\leq\) ACC\(^0\) \(\circ\) THR.

Building on [34], [17] recently generalized the connection to the average-case, by showing that strong enough circuit-analysis algorithms also imply \((1/2 + o(1))-inapproximability average-case lower bounds for NQP or NP. In particular, it was shown that NQP cannot be \((1/2 + 1/polylog(n))-approximated by ACC\(^0\) \(\circ\) THR. This is very interesting for two reasons: first, average-case lower bounds tend to have other applications such as constructing unconditional PRGs; second, the proof techniques do not apply the easy-witness lemma of [34, 49], and follows a more direct approach.

Still, the \((1/2 + 1/polylog(n))-inapproximability result is not enough to get us a non-trivial (say, with \(n^{o(1)}\) seed length) PRG construction for ACC\(^0\), which requires at least a \((1/2 + 1/n^{o(1)})\)-inapproximability bound.

The \(1/2 + 1/\sqrt{n}\) Razborov-Smolensky barrier. Indeed, proving a non-trivial \((1/2 + n^{-c(1)}))-inapproximability result is even open for AC\(^0\)[\(\mathbb{F}\)] circuits (AC\(^0\) circuits extended with parity gates). Using the renowned polynomial approximation method, [36, 41, 42] showed that the majority function cannot be \((1/2 + n^{1/2} - \varepsilon))-approximated by AC\(^0\)[\(\mathbb{F}\)]. However, it is even open that whether E\(^{NP}\) can be \((1/2 + 1/\sqrt{n})\)-approximated by \(\mathbb{F}_2\)-polynomials. Improving the \((1/2 + 1/\sqrt{n})\)-bound (and constructing the corresponding PRGs) is recognized as a significant open question in circuit complexity [16, 21, 43, 44].

1.2 Our Results

In this paper, we significantly improve the circuit-analysis-algorithm-to-average-case-lower-bounds connection in [17]. We first define the circuit-analysis task of our interest.

- CAPP\(^4\) for \(\mathcal{C}\) circuits with inverse-circuit-size error:
  Given a \(\mathcal{C}\) circuit \(C\) of size \(S\) on \(n\) input bits, estimate
  \[
  \Pr_{x \in \{0,1\}^n} [C(x) = 1]
  \]
  within an additive error \(1/S\).

For simplicity, throughout this paper, we will just refer to the above problem as CAPP. We remark that under the widely believed assumption PromiseBPP = PromiseP, this problem has a poly(S) time algorithm even for \(\mathcal{C} = \mathcal{P}/poly\). In the following, we show that indeed a non-trivial improvement on the brute-force \(2^n \cdot poly(S)\)-time algorithm already implies strong average-case lower bounds for \(\mathcal{C}\).

From Non-trivial CAPP Algorithms to Strong Average-Case Circuit Lower Bounds.

**Theorem 1.1.** Let \(\mathcal{C}\) be a typical circuit class\(^5\) such that \(\mathcal{C}\) circuits of size \(S\) can be implemented by (general) circuits of depth \(O\log(S)\).

The following hold.

1. **NP Average-Case Lower Bound** Suppose there is a constant \(\varepsilon > 0\) such that the CAPP problem of AND\(_4 \circ \mathcal{C}\) circuits of size \(2^n\) can be solved in \(2^{n-\varepsilon n}\) time. Then for every constant \(k \geq 1\), NP cannot be \((1/2 + n^{-k})\)-approximated by \(\mathcal{C}\) circuits of size \(n^k\).

2. **NQP Average-Case Lower Bound** Suppose there is a constant \(\varepsilon > 0\) such that the CAPP problem of AND\(_4 \circ \mathcal{C}\) circuits of size \(2^n\) can be solved in \(2^{n-\varepsilon n}\) time. Then for every constant \(k \geq 1\), NQP cannot be \((1/2 + 2^{-\log n})\)-approximated by \(\mathcal{C}\) circuits of size \(2^{\log n}\).

3. **NEXP Average-Case Lower Bound** Suppose the CAPP problem of AND\(_4 \circ \mathcal{C}\) circuits of size poly(n) can be solved in \(2^{n^{o(1)}}\) time. Then NE cannot be \((1/2 + 1/poly(n))\)-approximated by \(\mathcal{C}\) circuits of size poly(n).

By the standard Discriminator Lemma [27], we immediately obtain worst-case lower bounds for MA\(_\mathcal{C}\) \(\circ\) \(\mathcal{C}\) circuits as well.

**Corollary 1.2.** Under the algorithmic assumptions of Theorem 1.1, we obtain worst-case lower bounds for MA\(_\mathcal{C}\) \(\circ\) \(\mathcal{C}\) circuits in the corresponding cases: (1) NP not in \(n^{k-}\)-size MA\(_\mathcal{C}\) \(\circ\) \(\mathcal{C}\) for all \(k\); (2) NQP not in \(2^{\log n}\)-size MA\(_\mathcal{C}\) \(\circ\) \(\mathcal{C}\) for all \(k\); (3) NE not in poly(n)-size MA\(_\mathcal{C}\) \(\circ\) \(\mathcal{C}\).

**Remark 1.3.** We remark that the conclusions of Theorem 1.1 still hold if the corresponding CAPP algorithms are non-deterministic. That is, on any computational branch, it either outputs a correct estimation\(^6\) or rejects, and it does not reject all branches.

**Remark 1.4.** Theorem 1.1 assumes \(\mathcal{C}\) is a sub-class of NC\(^1\) (e.g., THR \(\circ\) THR, TC\(^0\), or ACC\(^0\)). On the other hand, if \(\mathcal{C}\) is stronger than NC\(^1\) (e.g., NC\(^0\), P\(_{\{poly\}}\)), [17, Theorem 1.3] already showed that\(^7\) even CAPP with constant error suffices to prove the stated average-case lower bounds in Theorem 1.1. Although we still left open the possible case that \(\mathcal{C}\) is incomparable to NC\(^1\), our theorem together with [17] cover nearly all interesting circuit classes.

**Comparison with [17].** Our Theorem 1.1 improves on the corresponding connection in [17] in two ways: (1) we get a much better inapproximability bound, which is crucial for our construction of nondeterministic PRGs; (2) we only need CAPP algorithms for AND\(_4 \circ \mathcal{C}\), while [17] requires algorithms for AC\(^0\) \(\circ\) \(\mathcal{C}\). On the other hand, our requirement on the CAPP algorithms is stronger (additive error 1/5) than that of [17] (constant additive error).

\(^{a}\)A circuit class \(\mathcal{C}\) is typical if it is closed under both negation and projection.

\(^{b}\)It is allowed that on different branches it outputs different estimations as long as they are all within an additive error of 1/S.

\(^{c}\)If \(\mathcal{C}\) is a typical circuit class and \(c = (1/2 + n^{-k})\) for some constant \(c\), then its proof can be generalized to the inapproximability corresponding to Theorem 1.1.

\(^{d}\)The acronym CAPP denotes the Circuit Acceptance Probability Problem.
More on our definition on CAPP. We remark that our definition of CAPP is a bit non-standard, comparing to the usual definition with a constant error. Nonetheless, such a CAPP algorithm is much weaker than a full-power #SAT algorithm, and (as discussed before) is widely believed to exist even for P/poly circuits.

Strong Average-Case Lower Bounds for ACC⁰ ⋄ THR. Applying the non-trivial #SAT algorithms for ACC⁰ ⋄ THR circuits in \[18\], it follows that NQP cannot be even weakly approximated by ACC⁰ ⋄ THR circuits, and it is (worst-case) hard for MAJ ⋄ ACC⁰ ⋄ THR circuits.

**Theorem 1.5.** For every constant \( k \geq 1 \), NQP cannot be \((1/2 + 2^{-\log^k n})\)-approximated by ACC⁰ ⋄ THR circuits of size \( 2^{\log^k n} \). Consequently, NQP cannot be computed by MAJ ⋄ ACC⁰ ⋄ THR circuits of size \( 2^{\log^k n} \) (in the worst-case), for all \( k \geq 1 \).

The same holds for \((\text{NP} \cap \text{coNP})_{/1}\) in place of NQP.

Non-deterministic PRGs for ACC⁰ with Sub-Polynomial Seed Length. As an important application of the above strong average-case lower bound, we also obtain the first PRG with \( n^{\log \log n} \) seed length for ACC⁰ circuits (previous, this was open even for \( \text{AC}^0[\theta] \) circuits), albeit it is non-deterministic and infinitely often.

**Theorem 1.6.** For every constant \( \epsilon > 0 \), there is an infinitely often, \((\text{NE} \cap \text{coNE})_{/1}\)-computable PRG fooling polynomial size ACC⁰ circuits with seed length \( 2^{(\log n)^\epsilon} \).

**Remark 1.7.** We can indeed optimize the seed length to be the inverse of any sub-fourth-exponential function. See [18, Section 7.2] for details.

Previously, the best PRG for ACC⁰ is from [20], which is \((\text{NE} \cap \text{coNE})_{/1}\)-computable and has seed length \( n - n^{1-\beta} \) for any constant \( \beta > 0 \). Our construction significantly improves on that.

**Lower Bounds and PRGs for Sum ⊕ Circuit.** For a circuit class \( \mathcal{C} \), a Sum ⊕ circuit is an \( \mathbb{F}_2 \)-linear combination \( C(x) := \sum_{i=1}^t \alpha_i C_i(x) \), such that each \( \alpha_i \in \mathbb{F}_2 \), each \( C_i \) is a \( \mathbb{F}_2 \) circuit on \( n \) input bits, and \( C(x) \in \{0,1\}^t \) for all \( x \in \{0,1\}^n \). We denote \( t \) as the sparsity of the circuit, and we define the size of \( C \) as the total size of all \( \mathcal{C} \) sub-circuits \( C_i \).

We first show that if we have the corresponding non-trivial #SAT algorithms instead of the non-trivial CAPP algorithms, we would have average-case lower bounds for Sum ⊕ circuits. To avoid repetition, in the following we only state the version for NQP.

**Corollary 1.8.** Let \( \mathcal{C} \) be a typical circuit class such that \( \mathcal{C} \) circuits of size \( S \) can be implemented by (general) circuits of depth \( O(\log S) \). Suppose there is a constant \( \epsilon > 0 \) such that the #SAT problem of AND₄ ⊕ \( \mathcal{C} \) circuits of size \( 2^{n^\epsilon} \) can be solved in \( 2^{n(1-\epsilon)\cdot n} \) time. Then for every constant \( k \geq 1 \), NQP cannot be \((1/2 + 2^{-\log^k n})\)-approximated by Sum ⊕ \( \mathcal{C} \) circuits of size \( 2^{\log^k n} \) size.

This immediately implies a strong average-case lower bound for Sum ⋄ ACC⁰ ⋄ THR.

**Corollary 1.9.** For every constant \( k \geq 1 \), NQP cannot be \((1/2 + 2^{-\log^k n})\)-approximated by Sum ⋄ ACC⁰ ⋄ THR circuits of size \( 2^{\log^k n} \). Consequently, NQP cannot be computed by MAJ ⋄ Sum ⋄ ACC⁰ ⋄ THR circuits of size \( 2^{\log^k n} \) (in the worst-case), for all \( k \geq 1 \).

The same holds for \((\text{NP} \cap \text{coNP})_{/1}\) in place of NQP.

Now we discuss some applications of our new techniques to some sub-classes of TC⁰ circuits.

We begin with some notation. Recall that a degree-\( d \) PTF gate is a function defined by \( \text{sign}(p(x)) \), where \( p \) is a degree-\( d \) polynomial on \( x \) over \( \mathbb{R} \), and \( \text{sign}(z) \) outputs \( 1 \) if \( z \geq 0 \) and \( 0 \) otherwise. Clearly, a THR gate is simply a degree-1 PTF gate.

[51] proved that NQP cannot be computed by \( n^k \)-size Sum ⋄ THR circuits for all \( k > 0 \). With our improved connection, we apply the #SAT algorithm for AND₄ ⋄ THR of [51] to improve it to a corresponding average-case lower bound.

**Theorem 1.10.** For all constants \( k \), NQP cannot be \((1/2 + 1/n^k)\)-approximated by \( n^k \)-size Sum ⋄ THR circuits. Consequently, NQP cannot be computed by \( n^k \)-size MAJ ⋄ Sum ⋄ THR circuits for all constants \( k \).

We remark that MAJ ⋄ Sum ⋄ THR is a sub-class of THR ⋄ THR with no previous known lower bounds. So Theorem 1.10 can be viewed as progress toward resolving the notorious open question of proving super-polynomial THR ⋄ THR lower bounds.

Applying the non-trivial zero-error #SAT algorithm for PTF in [10], we also obtain NQP (NE) average-case lower bounds for Sum ⋄ PTF₁ d circuits.

**Theorem 1.11.** The following hold.

- For every constants \( d, k \geq 1 \), NQP cannot be \((1/2 + 2^{-\log^k n})\)-approximated by Sum ⋄ PTF₁ d circuits of sparsity \( 2^{\log^k n} \). Consequently, NQP does not have \( 2^{\log^k n} \)-size MAJ ⋄ Sum ⋄ PTF₁ d circuits.
- Let \( d(n) = 0.49 \log n / \log \log n \) then NE cannot be \((1/2 + 1/poly(n))\)-approximated by Sum ⋄ PTF₁ d(n) circuits of sparsity \( poly(n) \). Consequently, NE \( \not\subseteq \text{MAJ} \cap \text{Sum} \cap \text{PTF₁ d(n)} \).

From the above theorem, we can also obtain non-trivial non-deterministic PRGs for Sum ⋄ PTF circuits.

**Theorem 1.12.** For every constants \( d, k \geq 1\) and \( \epsilon > 0 \), there is an \((\text{NE} \cap \text{coNE})_{/1}\)-computable i.o. PFG with seed length \( O(\log n \cdot 2^{O(d)}) \) [33]. Our construction has a worse seed-length, is non-deterministic and infinitely often, but works for the larger class Sum ⋄ PTF.

Towards TC⁰ Lower Bounds. In [19], it is shown that non-trivial CAPP algorithms for MAJ ⋄ PTF circuits with inverse-polynomial additive error would already imply PTF ⋄ THR circuit lower bounds. We significantly improve that connection by showing it would indeed imply TC⁰ lower bounds!

---

\( ^8 \) This average-case lower bound can also be extended to against Sum ⋄ ReLU circuits, similar to the exact Sum ⋄ ReLU lower bounds in [51].

\( ^9 \) We did not attempt to optimize this seed length.
Theorem 1.13. If there is a $2^n/n^{o(1)}$ time CAPP algorithm for poly(n)-size MAJ ⊕ MAJ circuits. Then NEXP ⊇ MAJ ⊕ MAJ ⊕ MAJ.

We remark that MAJ ⊕ MAJ ⊕ MAJ is actually equivalent to MAJ ⊕ THR ⊕ THR (since MAJ ⊕ MAJ = MAJ ⊕ THR [23]). Since exponential-size (worst-case) lower bounds against MAJ ⊕ MAJ are already known. If only we can “mine” a non-trivial CAPP algorithm (which is widely believed to exist) for MAJ ⊕ MAJ circuits from these lower bounds, we would have worst-case lower bounds against TC03.

Concurrent Works. A concurrent work by Viola [45] proved that E[NP] cannot be (1/2+log(k)/n)-approximated by ACC[0] circuits of size s and depth h. This result is incomparable with ours. We proved that E[NP] cannot be (1/2+ε)-approximated by ACC0 circuits of polynomial size for some $ε<1/n$, while the inapproximability result in [45] only achieves $ε>1/n$. On the other hand, our paper does not prove anything about exponential (e.g., $2^{n^{0.5}}$) sized ACC[0] circuits, while the results in [45] bypass the (1/2 + 1/√n)-barrier.

1.3 Intuition

In the following, we sketch the intuitions for our new average-case lower bounds.

In this section, we will aim for a simpler version that NQP cannot be (1/2 + $n^{-k}$)-approximated by ACC0 for a large constant $k$ (say, $k=10^5$) for simplicity. We believe this version already captures all important technical ideas of our new average-case circuit lower bounds.

1.3.1 Review of [17] and the Bottleneck. First, since our work crucially builds on [17] (which proved NQP cannot be (1/2+1/polylog(n))-approximated by ACC0), it would be very instructive to review the proof structure of [17], and understand what is the bottleneck of extending [17] to prove a (1/2 + $n^{-k}$)-inapproximability bound.

A high-level overview of [17]: three steps. Suppose we are proving NQP cannot be (1 − $δ$)-approximated by ACC0 for now, where $δ$ is a small constant. On a very high level, the proof of [17] involves the following three steps.12

Step I (Conditional collapse from NC1 to ACC0.)
Assuming NQP can be (1 − $δ$)-approximated by ACC0, [17] shows that NC1 collapses to ACC0, using the existence of self-reducible NC1-complete languages [9, 12, 31].

Step II (An NE algorithm certifying low depth hardness.)
Next, making use of the non-trivial SAT algorithm for ACC0 circuits [49], [17] shows that there is an NE algorithm V(·, ·) certifying $n^δ$-depth hardness. Formally, V(x, y) takes inputs such that |y| = $2^{x^δ}$; for infinitely many n’s, V(1n, ·) is satisfiable, and V(1n, y) = 1 implies y, interpreted as a function $f_y : [0, 1]^n → [0, 1]$, does not have $n^δ$-depth circuits.

Step III (Certifying low depth hardness implies average-case lower bounds for low depth circuits.)

Finally, [17] shows that the above algorithm V would be sufficient to imply that NQP cannot be (1 − $δ$)-approximated by NC1 (and also ACC0).

The bottleneck of the argument: Step I. Suppose we are going to prove NQP cannot be (1/2 + $n^{-k}$)-approximated by ACC0, let us examine which one of the above three steps would break.

Clearly, Step II is unaffected (assuming Step I works). Another observation is that since NC1 can compute majority [12], we can use the XOR Lemma [24, 29, 52] to show that NQP cannot be (1/2+$n^{-k}$)-approximated by NC1 circuits. Therefore, Step III still works if we want to prove the stronger (1/2 + $n^{-k}$)-inapproximability result.

However, Step I completely breaks. Assuming NQP can be (1/2 + $n^{-k}$)-approximated by ACC0 circuits, it seems hopeless to show that NC1 collapse to ACC0 using some random self-reducible languages. This is because the given circuit only (1/2 + $n^{-k}$)-approximates the given random self-reducible language, and to the best of our knowledge, all known corrector for such languages in this error regime requires computing at least some variants of the majority function, while ACC0 is conjectured not to be able to compute majority [41]!

1.3.2 A Detour: Chen and Williams [15] and $\overline{\text{SUM}}_δ \circ \text{ACC}$ Circuit Lower Bounds. So it seems unlikely that we can show a collapse theorem from NC1 to ACC0 under the assumption that NQP can be (1/2 + $n^{-k}$)-approximated by ACC0. A natural idea to avoid this obstacle is to show NC1 collapses to some other larger classes under the same assumption. Examining the proof idea of [17], it seems at least we can show NC1 collapses to MAJ ⊕ ACC0 under the assumption. However, the issue is that then we don’t know how to implement Step II, as we don’t have a non-trivial SAT (or even Gap-UNSAT) algorithm for MAJ ⊕ ACC0 circuits.

So we indeed want a collapse theorem which would collapse NC1 to a circuit class $\mathcal{C}$ for which we at least know some non-trivial algorithms for, and of course $\mathcal{C}$ also has to contain ACC0. Perhaps the best choice for us is the $\overline{\text{SUM}}_δ \circ \text{ACC}$ circuits which has recently been studied by [19]. So let us take a detour into this circuit class and the corresponding lower bounds in [19].

$\overline{\text{SUM}}_δ \circ \text{ACC}$ Circuits. Let $\mathcal{C}$ be a class of functions from $\{0, 1\}^n → \{0, 1\}$ and $δ \in [0, 0.5)$. We say $f : \{0, 1\}^n → \{0, 1\}$ admits a $\overline{\text{SUM}}_δ \circ \mathcal{C}$ circuit of sparsity S, if there are $S$ functions $C_1, C_2, \ldots, C_S$ from $\mathcal{C}$, together with $S$ coefficients $a_1, a_2, \ldots, a_S$, in $\mathbb{R}$, such that for all $x ∈ \{0, 1\}^n$,

$$\sum_{i=1}^{S} a_i \cdot C_i(x) = f(x) \leq δ.$$ 

Given a valid $\overline{\text{SUM}}_δ \circ \text{ACC}$ circuit $C$, we say $C(x) = 1$ if the corresponding output value $|\sum_{i} a_i C_i(x) - 1|$ ≤ $δ$, and $C(x) = 0$ otherwise. [19] gives a $2^{n^{0.5}}$-time Gap-UNSAT (in fact, constant-error CAPP) algorithm for $\overline{\text{SUM}}_δ \circ \text{ACC}$ of $2^{n^{0.5}}$-size when $δ$ is small (the algorithm is indeed already implicit in [51]). Building on this algorithm (and more importantly, PCP of proximity), [19] proves that NQP $∉ \overline{\text{SUM}}_δ \circ \text{ACC}$ for any constant $δ \in [0, 1/2)$.

11Actually, in [17], Step III is much more complicated than the previous two steps, and Step II just follows from [50]. In the presentation of [17], Step III is decomposed into several sub-steps [17, Section 6.2, 7-9]. We choose to give the overview in this way because we essentially make use of Step III as a black box, and our improvement is mostly focusing on the first two steps. In particular, our improved Step II is much more involved than that of [17], and crucially builds on [19].

12It is proved that black-box hardness amplification requires majority [26, 39].

13Precisely speaking, we have to start with our (N/1coN)QPj lower bounds for that purpose.
1.3.3 **Key Technical Ingredient**: A \(\oplus\)L-complete Language CMD with a \(\sum_\delta\) Error Corrector. So given the result of [19], the question becomes:

**A New Collapse Theorem?** Can we show a collapse from \(\text{NC}^1\) to \(\sum_2 \circ \text{ACC}^0\) circuits, assuming \(\text{NQP}\) can be \((1/2 + n^{-k})\)-approximated by \(\text{ACC}^0\) circuits?

Our improvement of Step I answers the question affirmatively, by making use of a \(\oplus\)L-complete\(^{14}\) language CMD [5, 25, 30] with very nice reducibility properties. We remark that the underlying techniques play a crucial part in the famous construction of \(\text{NC}^0\)-computable one-way functions (and low-stretch PRGs) [5] (see also the book [4]).

(1) \((\oplus\text{-completeness under projections). That is, for every language} L \in \oplus\text{L, there is a polynomial-time computable projection} F \text{ such that} L(x) = \text{CMD}(P(x)).\)

(2) \((\text{Single-query error correctability with a randomized image CMD.) For technical reasons, we also have to introduce another} \oplus\text{L-complete language CMD, which is a “randomized image” of CMD under projections (when randomness is fixed) [25, Claim 2.19]. That is, given} n \in \mathbb{N}, \text{there is} m = \text{poly}(n) \text{and a randomized reduction} P(x, r) \text{ (} r \text{ is the random bits) from CMD on input length} n \text{ to CMD on input length} m, \text{ such that:}\)

(a) \(\text{For all} x \in \{0, 1\}^n, P(x, U_r) \text{ distributes uniformly on} \{\{0, 1\}^m, \text{ where} \ell \text{ is the number of random bits involved, and} U_r \text{ is the uniform distribution over} \{0, 1\}^{\ell}.\)

(b) \(\text{For all fixed random bits} r, P(x, r) \text{ is a projection of} x.\)

(c) \(\text{For all} x \in \{0, 1\}^n, \text{CMD}_r(x) = \text{DCMD}_m(P(x, r)) \oplus r_0 \text{ for all} r, \text{ where} r_0 \text{ is the first bit of} r.\)

An error corrector in \(\sum_2 \circ f\). The second property of CMD stated above is amazing. It enables us to do the desired error correction with \(\sum_2 \circ f\) circuits (a linear sum of \(f\) functions composed with projections). See [18, Section 3] for the details. It then follows that if \(\text{NQP}\) can be \((1/2 + n^{-k})\)-approximated by \(\text{ACC}^0\) circuits, we would have the desired collapse from \(\text{NC}^1\) to \(\sum_2 \circ \text{ACC}^0\).

1.3.4 **A Simpler Proof for a Worst-Case Lower Bound Against MAJ \(\circ\) ACC\(^0\).** With the improved collapse result, we can already prove worst-case lower bounds against MAJ \(\circ\) ACC\(^0\). For simplicity, here we only show the following weaker version.

**Theorem 1.14 (Toy Example).** \(\text{NQP} \not\subseteq \text{MAJ} \circ \text{ACC}^0.\)

**Proof Sketch.** There are two cases.

- **First,** we assume \(\text{DCMD}\) (which is in \(\text{NQP}\)) cannot be \((1/2 + 1/\text{poly}(n))\)-approximated by \(\text{ACC}^0\). This implies that \(\text{NQP} \not\subseteq \text{MAJ} \circ \text{ACC}^0\), via the standard Discriminator Lemma [27].

- **Second,** suppose \(\text{DCMD}\) can be \((1/2 + 1/n^k)\)-approximated by \(n^k\)-size \(\text{ACC}^0\) circuits for a constant \(k\). This implies that \(\text{NC}^1\) collapses to \(\sum_2 \circ \text{ACC}^0\). By [19], \(\text{NQP} \not\subseteq \text{ACC}^0\). This in turn implies that \(\text{NQP} \not\subseteq \text{NC}^1\), and clearly also \(\text{NQP} \not\subseteq \text{MAJ} \circ \text{ACC}^0\). \(\square\)

**1.3.5 Toward Average-Case Hardness: The Updated Three Steps Plan.** Now we switch to the new average-case circuit lower bounds. With the new conditional collapse theorem, the following are our updated three steps plan for the new average-case lower bounds.\(^{15}\)

**Step I** (Conditional collapse from \(\text{NC}^1\) to \(\sum_2 \circ \text{ACC}^0\).)

Assuming \(\text{NQP}\) can be \((1/2 + n^{-k})\)-approximated by \(\text{ACC}^0\), we show that \(\text{NC}^1\) collapses to \(\sum_2 \circ \text{ACC}^0\), utilizing the nice properties of the problems CMD and DCMD.

**Step II** (An NE algorithm certifying low depth hardness.) Next, making use of the non-trivial constant error CAPP algorithm for \(\sum_2 \circ \text{ACC}^0\) circuits [19, 51], we show that there is an NE algorithm \(V(\cdot, \cdot)\) certifying \(n^\ell\)-depth hardness.

**Step III** (Certifying low depth hardness implies average-case lower bounds for low depth circuits.) Finally, we show that the above algorithm \(V\) would be sufficient to imply that \(\text{NQP}\) cannot be \((1/2 + n^{-k})\)-approximated by \(\text{NC}^1\) (and also \(\text{ACC}^0\)).

As previously discussed, Step III can be achieved easily by combining [17] and the XOR Lemma [24, 29, 52]. It remains to implement Step II, which is the most technical part of this work.

1.3.6 **Review of Step II: Certifying Hardness via PCP and Nondeterministic Time Hierarchy.** To implement Step II, the natural idea is to directly modify Step II ([17, Section 6.1]), and follow [50]. Now we briefly review the details of Step II and explain why it seems hard to adapt it directly.

**Setting up the verifier \(V_{\text{cert}}\).** Let \(L\) be a unary language in \(\text{NTIME}[2^n]\setminus\text{NTIME}[2^n/\text{poly}(n)]\) [54]. Fix an efficient PCP verifier \(V_{\text{PCP}}\) for \(L\) (such as [13]). That is, for a function \(\ell := \ell(n) = n + \text{O}(\log n), V_{\text{PCP}}(1^n)\) takes \(\ell\) random bits as input, runs in \(\text{poly}(n)\) time, is given access to an oracle \(O : \{0, 1\}^\ell \rightarrow \{0, 1\}\), and satisfies the following conditions:

1. (Completeness) if \(1^n \in L\), then there exists an oracle \(O\) such that \(V_{\text{PCP}}(1^n, O)\) always accepts;

2. (Soundness) if \(1^n \notin L\), then for all oracles \(O\), the probability \(V_{\text{PCP}}(1^n, O)\) accepts is \(\leq 1/\text{poly}(n)\).

Now, we define \(V_{\text{cert}}\) as follows:\(^{15}\) \(V_{\text{cert}}(1^n, y)\) treats \(y\) as the truth-table of an oracle \(O_y : \{0, 1\}^\ell \rightarrow \{0, 1\}\), and verifies whether \(V_{\text{PCP}}(1^n, O_y)\) always accepts\(^{15}\). Clearly, \(V_{\text{cert}}\) runs in \(\text{poly}(n + |y|)\) time.

Since any depth-\(d\) circuit is equivalent to some \(2^{O(d)}\)-size \(\sum_2 \circ \text{ACC}^0\) circuit (recall that now \(\text{NC}^1\) collapses to \(\sum_2 \circ \text{ACC}^0\)), to show that \(V_{\text{cert}}\) certifies \(n^\ell\)-depth hardness, it suffices to show that \(V_{\text{cert}}\) certifies hardness for \(2^{n^\ell}\)-size \(\sum_2 \circ \text{ACC}^0\) circuits for \(r > c_1\).

Let us suppose the opposite that \(V_{\text{cert}}\) does not certify hardness for \(2^{n^\ell}\)-size \(\sum_2 \circ \text{ACC}^0\) circuits. In particular, this means for all large enough \(n\), if \(V_{\text{cert}}(1^n, \cdot)\) is satisfiable, then there is a \(2^{n^\ell}\)-size \(\sum_2 \circ \text{ACC}^0\) circuit \(C\) such that \(V_{\text{cert}}(1^n, t(C)) = 1\), where \(t(C)\) is the truth-table of \(C\). Translating it to the setting of PCP, for large enough \(n\), the following hold:

1. (Succinct Completeness) if \(1^n \in L\), then there exists a \(2^{n^\ell}\)-size \(\sum_2 \circ \text{ACC}^0\) circuit \(C : \{0, 1\}^\ell \rightarrow \{0, 1\}\) such that \(V_{\text{PCP}}(1^n, C)\) always accepts.

\(^{14}\)Roughly speaking, \(\oplus\)L consists of languages \(L\) such that there is an \(O(\log n)\) space nondeterministic Turing machine \(M\), such that on every input \(x, x \in L\) if and only if there is an odd number of computational paths making \(M\) accept on input \(x\).

\(^{15}\)Strictly speaking, here \(|y| = 2^\ell = 2^{n^\ell} \cdot \text{poly}(n)\) which is slightly larger than \(2^n\), but this slight difference does not really matter in the proof.
Let $D_C := V_{PCP}(1^n)^C$. We would like to accept when $\hat{p}_{acc}(D_C) = 1$, and reject when $\hat{p}_{acc}(D_C) < 1/n$, so a constant additive error (say, 1/10) approximation to $\hat{p}_{acc}(D_C)$ would suffice.

The issue here is that, $D_C$ is not a $\sum_3 \circ ACC^0$ circuit anymore. So we don’t know how to estimate $\hat{p}_{acc}(D_C)$ using the constant error PCPP algorithm for $\sum_3 \circ ACC^0$ in [19, 51].

We remark that by [13], $V_{PCP}$ can indeed be implemented by a 3-CNF, hence if $C$ is only an $ACC^0$ circuit, $V_{PCP}(1^n)^C$ is still an $ACC^0$ circuit. This is why this argument works in the original Step II, where we have a collapse from $NC^1$ to $ACC^0$ instead of $\sum_3 \circ ACC^0$.

### 1.3.7 Getting Around of the Issue with PCP of Proximity

To avoid the aforementioned issue, we would like to adopt the PCP of Proximity framework introduced in [19], which also plays a crucial part in the $p^{\#}$ construction of rigid matrices in [3]. For more intuition on this framework and how it compares to and improves on the earlier works [47, 49], one is referred to [19, Section 1.6].

For a SAT instance $F$, $Y$ a subset of its variables, and $y \in \{0, 1\}^{|Y|}$, we use $F^{\bot}_Y$ to denote the resulting instance obtained by assigning the $Y$ variables in $F$ to $y$. We also use $OPT(F)$ to denote the maximum fraction of clauses that can be satisfied by any assignment.

The following transformation is the key technical part of [19].

**Theorem 1.15 (Implicit in [19]).** Let $Enc$ be the encoder of some constant-rate error correcting code. There is a polynomial-time transformation that, given a circuit $D$ on $n$ inputs of size $m \geq n$, outputs a 2-SAT instance $F$ on variable set $Y \cup Z$, where $|Y| = O(n)$, $|Z| \leq poly(m)$ and $F$ has $poly(m)$ clauses, such that for two constants $c_{PCPP} > sp_{PCPP}$, the following hold for all $x \in \{0, 1\}^n$.

- If $D(x) = 1$, then $OPT(F_{Y = Enc(x)}) \geq c_{PCPP}$. Furthermore, there is a poly(m)-time algorithm computing a corresponding $z_D(x)$ given $x$ which satisfies at least a $c_{PCPP}$ fraction of clauses.
- If $D(x) = 0$, then $OPT(F_{Y = Enc(x)}) \leq sp_{PCPP}$.

The key idea of [19] is to apply the above transformation on the obtained circuit $D_C$, and guess the corresponding $\mathcal{C}$ circuits for each output bit of the function $z_{D_C}(x)$. In [19], the focus is to prove worst-case lower bounds like NQP $\subseteq \mathcal{C}$ for a circuit class $\mathcal{C}$. Therefore, we can safely assume $P \subseteq \mathcal{C}$ and there exist corresponding $\mathcal{C}$ circuits for each output bit of $z_{D_C}(x)$.

---

18Note that here we are waiving the very important issue of how to test whether the guessed $\sum_3 \circ ACC^0$ is valid. We will discuss this issue at the end of the section.

19Here we don’t remove the already satisfied clauses or the clauses which cannot be satisfied after the partial assignment.

20This formulation is due to [46].
Checking the guessed $\widetilde{\text{Sum}}_\delta \circ \text{ACC}^0$ circuits. Finally, as we have remarked briefly before, we waived an important issue on checking whether the guessed $\text{Sum}_\delta \circ \text{ACC}^0$ circuits are valid (that is, whether the linear sum is close to either 0 or 1 on all inputs $x$). This is because in the algorithm described above, when $x \notin L$, it is still possible that we guess some invalid $\text{Sum}_\delta \circ \text{ACC}^0$ circuits $T_1, T_2, \ldots, T_{|Z|}$ and conclude that $p_{\text{key}} > \frac{\text{PCPP} + \text{PCPP}}{2}$, as the constant error CAPP algorithm for $\text{Sum}_\delta \circ \text{ACC}^0$ may behave arbitrarily on invalid $\text{Sum}_\delta \circ \text{ACC}^0$ circuits.

More formally, given a presumed $\text{Sum}_\delta \circ \text{ACC}^0$ circuit $C$, let $f(x)$ be the corresponding $\sum_i a_i C_i(x)$, and

$$\text{bin}_f(x) := \begin{cases} 1 & f(x) > 1/2, \\ 0 & \text{otherwise}. \end{cases}$$

To test whether $C$ is valid, we want to check whether $\|\text{bin}_f - \frac{f}{\|f\|_\infty}\|_\infty \leq \delta$. Ideally, we want a test which accepts when $\|\text{bin}_f - \frac{f}{\|f\|_\infty}\|_\infty \leq \delta$ and rejects when (say) $\|\text{bin}_f - \frac{f}{\|f\|_\infty}\|_\infty \geq 3\delta$. But this turns out to be too hard.

Luckily, a careful examination shows that we only have to reject when $\|\text{bin}_f - \frac{f}{\|f\|_\infty}\|_2 \geq 3\delta$, and this can be solved by a careful polynomial manipulation as in [19]. See [18, Section 5] for the details.

2 OPEN PROBLEMS

We conclude with several interesting open problems stemming from our work.

1. The most exciting open question would be to apply Theorem 1.13 to prove super-polynomial lower bounds for $\text{TC}_0$.

2. Are there $\text{P}$-complete problems with similar random reducibility properties of CMD and DCMD? Besides being an interesting problem in its own right, the existence of such a problem would greatly simplify our framework for strong average-case lower bounds. In particular, we will no longer need hard MA problems with low depth predicates, and PCP with low depth computable proofs.

3. The seed length of our i.o. NPRG fooling $\text{ACC}^0$ circuits is only inverse sub-half-exponential. Can we obtain an i.o. NPRG with polylog$(n)$ seed length? As a related question, can we show that there is a constant $\varepsilon > 0$ such that $\text{E}^{\text{NP}}$ cannot be $(1/2 + 1/2^{n^{\varepsilon}})$-approximated by $\text{ACC}^0$ circuits of $2^{n^{\varepsilon}}$ size? This paper only implicitly proves that $\text{E}^{\text{NP}}$ cannot be $(1/2 + 1/|f(n)|)$-approximated by $\text{ACC}^0$ circuits of $|f(n)|$ size for sub-half-exponential $f(n)$.

4. Since we have proved lower bounds for MA \circ \text{ACC}^0, the natural next step would be to prove lower bounds for THR \circ \text{ACC}^0. Can we formulate any algorithmic approach to prove such a lower bound? That is, are there certain non-trivial circuit-analysis algorithms for $\mathcal{C}$ which would imply THR \circ $\mathcal{C}$ lower bounds?

It seems plausible to us that non-trivial \#SAT algorithms would suffice (note that we already proved non-trivial \#SAT algorithms for $\mathcal{C}$ imply MA \circ $\mathcal{C}$ lower bounds, which is a non-trivial sub-class of THR \circ $\mathcal{C}$). Such a connection would also imply lower bounds for THR \circ \text{ACC}^0 \circ THR, which is (much) stronger than the already notorious circuit class THR \circ THR.

5. Is THR contained in MA \circ \text{ACC}^0? (Or even MA \circ $\text{Sum}$ \circ \text{ACC}^0?) We don’t have an inclination on the answer. But if it is contained in MA \circ \text{ACC}^0, it would immediately imply super-polynomial lower bounds for THR \circ THR.

6. Vyas and Williams [46] conjectured that $\text{SYM} \circ \mathcal{C}$ lower bounds should follow from \#SAT algorithms for $\mathcal{C}$, where $\text{SYM}$ denotes arbitrary symmetric functions. Can the new techniques in this paper help to prove this conjecture?

ACKNOWLEDGMENTS

The first author wants to thank his advisor Ryan Williams for basically everything: introducing the problem of proving lower bounds for MA \circ \text{ACC}^0 to him and being very optimistic about its resolution, countless encouraging and valuable discussion during the project, and also suggestions and comments on an early draft of this paper. The second author also wants to thank Ryan Williams for hosting him during the summer of 2019 and many inspiring discussions during the period.

We would like to thank Nikhil Vyas for helpful discussions.

REFERENCES


