The Acrobatics of BQP

Scott Aaronson
University of Texas at Austin, TX, USA

DeVon Ingram
University of Chicago, IL, USA

William Kretschmer
University of Texas at Austin, TX, USA

Abstract

One can fix the randomness used by a randomized algorithm, but there is no analogous notion of fixing the quantumness used by a quantum algorithm. Underscoring this fundamental difference, we show that, in the black-box setting, the behavior of quantum polynomial-time (BQP) can be remarkably decoupled from that of classical complexity classes like NP. Specifically:

- There exists an oracle relative to which $\mathsf{NP}^{\mathsf{BQP}} \not\subset \mathsf{BQP}^{\mathsf{PH}}$, resolving a 2005 problem of Fortnow. As a corollary, there exists an oracle relative to which $\mathsf{P} = \mathsf{NP}$ but $\mathsf{BQP} \neq \mathsf{QCMA}$.
- Conversely, there exists an oracle relative to which $\mathsf{BQP}^{\mathsf{NP}} \not\subset \mathsf{PH}^{\mathsf{BQP}}$.
- Relative to a random oracle, PP is not contained in the "QMA hierarchy" $\mathsf{QMA}^{\mathsf{QMA}^{\mathsf{QMA}^{\cdots}}}$.
- Relative to a random oracle, $\Sigma_{k+1}^{\mathsf{P}} \not\subset \mathsf{BQP}^{\Sigma_k^{\mathsf{P}}}$ for every k.
- There exists an oracle relative to which $\mathsf{BQP} = \mathsf{P}^{\#\mathsf{P}}$ and yet PH is infinite. (By contrast, relative to all oracles, if $\mathsf{NP} \subseteq \mathsf{BPP}$, then PH collapses.)
- There exists an oracle relative to which $\mathsf{P} = \mathsf{NP} \neq \mathsf{BQP} = \mathsf{P}^{\#\mathsf{P}}$.

To achieve these results, we build on the 2018 achievement by Raz and Tal of an oracle relative to which $\mathsf{BQP} \not\subset \mathsf{PH}$, and associated results about the FORRELATION problem. We also introduce new tools that might be of independent interest. These include a "quantum-aware" version of the random restriction method, a concentration theorem for the block sensitivity of AC^0 circuits, and a (provable) analogue of the Aaronson-Ambainis Conjecture for sparse oracles.

2012 ACM Subject Classification Theory of computation → Quantum complexity theory; Theory of computation → Complexity classes

Keywords and phrases BQP, Forrelation, oracle separations, Polynomial Hierarchy, query complexity

Digital Object Identifier 10.4230/LIPIcs.CCC.2022.20

Related Version *Full Version*: `https://arxiv.org/abs/2111.10409` [8]

Funding *Scott Aaronson*: Supported by a Vannevar Bush Fellowship from the US Department of Defense, a Simons Investigator Award, and the Simons "It from Qubit" collaboration.
DeVon Ingram: Supported by an NSF Graduate Research Fellowship.
William Kretschmer: Supported by an NDSEG Fellowship.

Acknowledgements We thank Lance Fortnow, Greg Kuperberg, Patrick Rall, and Avishay Tal for helpful conversations. We are especially grateful to Avishay Tal for helping us prove a tail bound on the block sensitivity of AC^0 circuits.

1 Introduction

The complexity-theoretic study of quantum computation is often dated from 1993, when Bernstein and Vazirani [15] defined BQP, or Bounded-Error Quantum Polynomial-Time: the class of languages that admit efficient quantum algorithms. Then as now, a central concern was how BQP relates to classical complexity classes, such as P, NP, and PH. Among the countless questions that one could raise here, let us single out three as especially fundamental:

37th Computational Complexity Conference (CCC 2022).
Editor: Shachar Lovett; Article No. 20; pp. 20:1–20:17
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

(1) Can quantum computers efficiently solve any problems that classical computers cannot? In other words, does $\mathsf{BPP} = \mathsf{BQP}$?
(2) Can quantum computers solve NP-complete problems in polynomial time? In other words, is $\mathsf{NP} \subseteq \mathsf{BQP}$?
(3) What is the best classical upper bound on the power of quantum computation? Is $\mathsf{BQP} \subseteq \mathsf{NP}$? Is $\mathsf{BQP} \subseteq \mathsf{PH}$?

Three decades later, all three of these still stand as defining questions of the field. Nevertheless, from the early 2000s onwards, it became rare for work in quantum computing theory to address any of these questions directly, perhaps simply because it became too hard to say anything new about them. A major recent exception was the seminal work of Raz and Tal [38], who gave an oracle relative to which $\mathsf{BQP} \not\subset \mathsf{PH}$, by completing a program proposed by one of us [2]. In this paper, we take the Raz-Tal breakthrough as a starting point. Using it, together with new tools that we develop, we manage to prove many new theorems about the power of BQP – at least in the black-box setting where much of our knowledge of quantum algorithms resides.

Before discussing the black-box setting or Raz-Tal, though, let's start by reviewing what is known in general about BQP. Bernstein and Vazirani [15] showed that $\mathsf{BPP} \subseteq \mathsf{BQP} \subseteq \mathsf{P}^{\#\mathsf{P}}$, and Adleman, DeMarrais, and Huang [10] improved the upper bound to $\mathsf{BQP} \subseteq \mathsf{PP}$, giving us the following chain of inclusions:

$$\mathsf{P} \subseteq \mathsf{BPP} \subseteq \mathsf{BQP} \subseteq \mathsf{PP} \subseteq \mathsf{P}^{\#\mathsf{P}} \subseteq \mathsf{PSPACE} \subseteq \mathsf{EXP}.$$

Fortnow and Rogers [21] slightly strengthened the inclusion $\mathsf{BQP} \subseteq \mathsf{PP}$, to show for example that $\mathsf{PP}^{\mathsf{BQP}} = \mathsf{PP}$. This complemented the result of Bennett, Bernstein, Brassard, and Vazirani [14] that $\mathsf{BQP}^{\mathsf{BQP}} = \mathsf{BQP}$: that is, BQP is "self-low," or "the BQP hierarchy collapses to BQP."

1.1 The Contrast with BPP

Meanwhile, though, the relationships between BQP and complexity classes like NP, PH, and $\mathsf{P/poly}$ have remained mysterious. Besides the fundamental questions mentioned above – is $\mathsf{NP} \subseteq \mathsf{BQP}$? is $\mathsf{BQP} \subseteq \mathsf{NP}$? is $\mathsf{BQP} \subseteq \mathsf{PH}$? – one could ask other questions:

(i) In a 2005 blog post, Fortnow [20] raised the question of whether $\mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{BQP}^{\mathsf{NP}}$. Do we even have $\mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{BQP}^{\mathsf{PH}}$? I.e., when quantum computation is combined with classical nondeterminism, how does the order of combination matter?
(ii) What about the converse: is $\mathsf{BQP}^{\mathsf{NP}} \subseteq \mathsf{PH}^{\mathsf{BQP}}$?
(iii) Suppose $\mathsf{NP} \subseteq \mathsf{BQP}$. Does it follow that $\mathsf{PH} \subseteq \mathsf{BQP}$ as well?
(iv) Suppose $\mathsf{NP} \subseteq \mathsf{BQP}$. Does it follow that PH collapses?
(v) Is $\mathsf{BQP} \subset \mathsf{P/poly}$?
(vi) Suppose $\mathsf{P} = \mathsf{NP}$. Does it follow that BQP is "small" (say, not equal to EXP)?
(vii) Suppose $\mathsf{P} = \mathsf{NP}$. Does it follow that $\mathsf{BQP} = \mathsf{QCMA}$, where QCMA (Quantum Classical Merlin Arthur) is the analogue of NP with a BQP verifier?

What is particularly noteworthy about the questions above is that, if we replace BQP by BPP, then positive answers are known to all of them:

(i) $\mathsf{NP}^{\mathsf{BPP}} \subseteq \mathsf{AM} \subseteq \mathsf{BPP}^{\mathsf{NP}}$.
(ii) $\mathsf{BPP}^{\mathsf{NP}} \subseteq \mathsf{PH} = \mathsf{PH}^{\mathsf{BPP}}$.
(iii) If $\mathsf{NP} \subseteq \mathsf{BPP}$, then $\mathsf{PH} = \mathsf{BPP}$ – this is sometimes given as a homework exercise in complexity theory courses, and also follows from (i).

(iv) If $\mathsf{NP} \subseteq \mathsf{BPP}$, then $\mathsf{PH} = \Sigma_2^{\mathsf{P}}$ – this follows from (iii) and the Sipser-Lautemann Theorem [45, 34].

(v) $\mathsf{BPP} \subset \mathsf{P/poly}$ is Adleman's Theorem [9].

(vi) If $\mathsf{P} = \mathsf{NP}$, then $\mathsf{P} = \mathsf{BPP}$ and hence $\mathsf{BPP} \neq \mathsf{EXP}$, by the time hierarchy theorem.

(vii) If $\mathsf{P} = \mathsf{NP}$, then of course $\mathsf{BPP} = \mathsf{MA}$.

So what is it that distinguishes BPP from BQP in these cases? In all of the above examples, the answer turns out to be one of the fundamental properties of classical randomized algorithms: namely, that one can always "pull the randomness out" from such algorithms, viewing them as simply deterministic algorithms that take a uniform random string r as an auxiliary input, in addition to their "main" input x. This, in turn, enables one to play all sorts of tricks with such an algorithm $M(x, r)$ – from using approximate counting to estimate the fraction of r's that cause $M(x, r)$ to accept, to moving r from inside to outside a quantifier, to hardwiring r as advice. By contrast, there is no analogous notion of "pulling the randomness (or quantumness) out of a quantum algorithm." In quantum computation, randomness is just an intrinsic part of the model that rears its head at the *end* (rather than the beginning) of a computation, when we take the squared absolute values of amplitudes to get probabilities.

This difference between randomized and quantum algorithms is crucial to the analysis of the so-called "sampling-based quantum supremacy experiments" – for example, those recently carried out by Google [11] and USTC [49]. The theoretical foundations of these experiments were laid a decade ago, in the work of Aaronson and Arkhipov [5] on BosonSampling, and (independently) Bremner, Jozsa, and Shepherd [19] on the commuting Hamiltonians or IQP model. Roughly speaking, the idea is that, by using a quantum computer, one can efficiently sample a probability distribution $\mathcal{D}$ over n-bit strings such that even *estimating* the probabilities of the outcomes is a #P-hard problem. Meanwhile, though, if there were a polynomial-time classical randomized algorithm $M(x, r)$ to sample from the same distribution $\mathcal{D}$, then one could use the "pulling out r" trick to estimate the probabilities of M's outcomes in PH. But this would put $\mathsf{P}^{\#\mathsf{P}}$ into PH, thereby collapsing PH by Toda's Theorem [47].

More generally, with any of the apparent differences between quantum algorithms and classical randomized algorithms, the question is: how can we prove that the difference is genuine, that no trick will ever be discovered that makes BQP behave more like BPP? For questions like whether $\mathsf{NP} \subseteq \mathsf{BQP}$ or whether $\mathsf{BQP} \subseteq \mathsf{NP}$, the hard truth here is that not only have we been unable to resolve these questions in the unrelativized world, we've been able to say little more about them than certain "obvious" implications. For example, suppose $\mathsf{NP} \subseteq \mathsf{BQP}$ *and* $\mathsf{BQP} \subseteq \mathsf{AM}$. Then since BQP is closed under complement, we would also have $\mathsf{coNP} \subseteq \mathsf{BQP}$, and hence $\mathsf{coNP} \subseteq \mathsf{AM}$, which is known to imply a collapse of PH [17]. And thus, if PH is infinite, then either $\mathsf{NP} \not\subseteq \mathsf{BQP}$ *or* $\mathsf{BQP} \not\subseteq \mathsf{AM}$. How can we say anything more interesting and nontrivial?

1.2 Relativization

Since the work of Baker, Gill, and Solovay [13], whenever complexity theorists were faced with an impasse like the one above, a central tool has been *relativized* or *black-box* complexity: in other words, studying what happens when all the complexity classes one cares about are fed some specially-constructed oracle. Much like perturbation theory in physics, relativization lets us make well-defined progress even when the original questions we wanted to answer are out of reach. It is well-known that relativization is an imperfect tool – the $\mathsf{IP} = \mathsf{PSPACE}$ [42], $\mathsf{MIP} = \mathsf{NEXP}$ [12], and more recently, $\mathsf{MIP}^* = \mathsf{RE}$ [30] theorems provide famous examples where complexity classes turned out to be equal, even in the teeth of oracles relative to which they were unequal. On the other hand, so far, almost all such examples have originated from a single source: namely, the use of algebraic techniques in interactive proof systems. And if, for example, we want to understand the consequences of $\mathsf{NP} \subseteq \mathsf{BQP}$, then arguably it makes little sense to search for nonrelativizing consequences if we don't even understand yet what the relativizing consequences (that is, the consequences that hold relative to all oracles) are or are not.

In quantum complexity theory, even more than in classical complexity theory, relativization has been an inextricable part of progress from the very beginning. The likely explanation is that, even when we just count queries to an oracle, in the quantum setting we need to consider algorithms that query all oracle bits in superposition – so that even in the most basic scenarios, it is already unintuitive what can and cannot be done, and so oracle results must do much more than formalize the obvious.

More concretely, Bernstein and Vazirani [15] introduced some of the basic techniques of quantum algorithms in order to prove, for the first time, that there exists an oracle A such that $\mathsf{BPP}^A \neq \mathsf{BQP}^A$. Shortly afterward, Simon [44] gave a quantitatively stronger oracle separation between BPP and BQP, and then Shor [43] gave a still stronger separation, along the way to his famous discovery that Factoring is in BQP.

On the negative side, Bennett, Bernstein, Brassard, and Vazirani [14] showed that there exists an oracle relative to which $\mathsf{NP} \not\subset \mathsf{BQP}$: indeed, relative to which there are problems that take n time for an NP machine but $\Omega\left(2^{n/2}\right)$ time for a BQP machine. Following the discovery of Grover's algorithm [25], which quantumly searches any list of N items in $O\left(\sqrt{N}\right)$ queries, the result of Bennett, Bernstein, Brassard, and Vazirani gained the interpretation that *Grover's algorithm is optimal.* In other words, any quantum algorithm for NP-complete problems that gets more than the square-root speedup of Grover's algorithm must be "non-black-box." It must exploit the structure of a particular NP-complete problem much like a classical algorithm would have to, rather than treating the problem as just an abstract space of 2^n possible solutions.

Meanwhile, clearly there are oracles relative to which $\mathsf{P} = \mathsf{BQP}$ – for example, a PSPACE-complete oracle. But we can ask: would such oracles necessarily collapse the hierarchy of classical complexity classes as well? In a prescient result that provided an early example of the sort of thing we do in this paper, Fortnow and Rogers [21] showed that there exists an oracle relative to which $\mathsf{P} = \mathsf{BQP}$ and yet PH is infinite. In other words, if $\mathsf{P} = \mathsf{BQP}$ would imply a collapse of the polynomial hierarchy, then it cannot be for a relativizing reason. Aaronson and Chen [6] extended this to show that there exists an oracle relative to which *sampling-based quantum supremacy is impossible* – i.e., any probability distribution approximately samplable in quantum polynomial time is also approximately samplable in classical polynomial time – and yet PH is infinite. In other words, if it is possible to prove the central theoretical conjecture of quantum supremacy – namely, that there are noisy quantum sampling experiments that cannot be simulated in classical polynomial time unless PH collapses – then nonrelativizing techniques will be needed there as well.

What about showing the power of BQP, by giving oracle obstructions to containments like $\mathsf{BQP} \subseteq \mathsf{NP}$, or $\mathsf{BQP} \subseteq \mathsf{PH}$? There, until recently, the progress was much more limited. Watrous [48] showed that there exists an oracle relative to which $\mathsf{BQP} \not\subset \mathsf{NP}$ and even $\mathsf{BQP} \not\subset \mathsf{MA}$ (these separations could also have been shown using the Recursive Fourier Sampling problem, introduced by Bernstein and Vazirani [15]). But extending this further, to get an oracle relative to which $\mathsf{BQP} \not\subset \mathsf{PH}$ or even $\mathsf{BQP} \not\subset \mathsf{AM}$, remained an open problem for two decades. Aaronson [2] proposed a program for proving an oracle separation between BQP and PH, involving a new problem he introduced called Forrelation:

▶ **Problem 1** (Forrelation). *Given black-box access to two Boolean functions $f, g : \{0,1\}^n \to \{1,-1\}$, and promised that either*

(i) *f and g are uniformly random and independent, or*

(ii) *f and g are uniformly random individually, but g has $\Omega(1)$ correlation with $\hat{f}$, the Boolean Fourier transform of f (i.e., f and g are "Forrelated"),*

decide which.

Aaronson [2] showed that Forrelation is solvable, with constant bias, using only a single quantum query to f and g (and $O(n)$ time). By contrast, he showed that any classical randomized algorithm for the problem needs $\Omega\left(2^{n/4}\right)$ queries – improved by Aaronson and Ambainis [4] to $\Omega\left(\frac{2^{n/2}}{n}\right)$ queries, which is essentially tight. The central conjecture, which Aaronson left open, said that Forrelation $\notin$ PH – or equivalently, by the connection between PH machines and AC^0 circuits [22], that there are no AC^0 circuits for Forrelation of constant depth and $2^{\mathrm{poly}(n)}$ size.

Finally, Raz and Tal [38] managed to prove Aaronson's conjecture, and thereby obtain the long-sought oracle separation between BQP and PH.[1] Raz and Tal achieved this by introducing new techniques for constant-depth circuit lower bounds, involving Brownian motion and the L_1-weight of the low-order Fourier coefficients of AC^0 functions. Relevantly for us, Raz and Tal actually proved the following stronger result:

▶ **Theorem 2** ([38]). *A PH machine can guess whether f and g are uniform or Forrelated with bias at most $2^{-\Omega(n)}$.*

Recall that before Raz and Tal, we did not even have an oracle relative to which $\mathsf{BQP} \not\subset \mathsf{AM}$. Notice that, if $\mathsf{BQP} \subseteq \mathsf{AM}$, then many other conclusions would follow in a relativizing way. For example, we would have:

- $\mathsf{P} = \mathsf{NP}$ implies $\mathsf{P} = \mathsf{BQP}$,
- $\mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{NP}^{\mathsf{AM} \cap \mathsf{coAM}} \subseteq \mathsf{BPP}^{\mathsf{NP}} \subseteq \mathsf{BQP}^{\mathsf{NP}}$,
- If $\mathsf{NP} \subseteq \mathsf{BQP}$, then $\mathsf{NP}^{\mathsf{NP}} \subseteq \mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{BQP}^{\mathsf{NP}} = \mathsf{BQP}^{\mathsf{BQP}} = \mathsf{BQP}$, and
- If $\mathsf{NP} \subseteq \mathsf{BQP}$, then $\mathsf{NP} \subseteq \mathsf{coAM}$, which implies that PH collapses.

Looking at it a different way, our inability even to separate BQP from AM by an oracle served as an obstruction to numerous other oracle separations.

The starting point of this paper was the following question: in a "post-Raz-Tal world," can we at last completely "unshackle" BQP from P, NP, and PH, by showing that there are no relativizing obstructions to any possible answers to questions like the ones we asked in Section 1.1?

1.3 Our Results

We achieve new oracle separations that show an astonishing range of possible behaviors for BQP and related complexity classes – in at least one case, resolving a longstanding open problem in this topic. Our title, "The Acrobatics of BQP," comes from a unifying theme of the new results being "freedom." We will show that, as far as relativizing techniques can detect, collapses and separations of classical complexity classes place surprisingly few constraints on the power of quantum computation. In most cases, this can be understood as ultimately

[1] Strictly speaking, they did this for a variant of Forrelation where the correlation between g and $\hat{f}$ is only $\sim \frac{1}{n}$, and thus a quantum algorithm needs $\sim n$ queries to solve the problem, but this will not affect anything that follows.

stemming from the fact that one cannot "fix the randomness" (or quantumness) used by a quantum algorithm, similarly to how one fixes the randomness used by a randomized algorithm in many complexity-theoretic arguments.

As we alluded to earlier, many of our new results would not have been possible without Raz and Tal's analysis of FORRELATION [38], which we rely on extensively. We will treat FORRELATION no longer as just an isolated problem, but as a sort of cryptographic code, by which an oracle can systematically make certain information available to BQP machines while keeping the information hidden from classical machines.

Having said that, very few of our results will follow from Raz-Tal in any straightforward way. Most often we need to develop other lower bound tools, in addition to or instead of Raz-Tal. Our new tools, which seem likely to be of independent interest, include a random restriction lemma for quantum query algorithms, a concentration theorem for the block sensitivity of AC^0 functions, and a provable analogue of the Aaronson-Ambainis conjecture [3] for certain sparse oracles.

Perhaps our single most interesting result is the following.

▶ **Theorem 3.** *There exists an oracle relative to which* $\mathsf{NP}^{\mathsf{BQP}} \not\subset \mathsf{BQP}^{\mathsf{NP}}$*, and indeed* $\mathsf{NP}^{\mathsf{BQP}} \not\subset \mathsf{BQP}^{\mathsf{PH}}$.

As mentioned earlier, Theorem 3 resolves an open problem of Fortnow [20], and demonstrates a clear difference between BPP and BQP that exemplifies the impossibility of pulling the randomness out of a quantum algorithm. Indeed, Theorem 3 shows that there is no general, black-box way to move quantumness past an NP quantifier, like we can do for classical randomness.

As a straightforward byproduct of Theorem 3, we are also able to prove the following:

▶ **Theorem 4.** *There exists an oracle relative to which* $\mathsf{P} = \mathsf{NP}$ *but* $\mathsf{BQP} \neq \mathsf{QCMA}$.

Conversely, it will follow from one of our later results, Theorem 9, that there exists an oracle relative to which $\mathsf{P} \neq \mathsf{NP}$ and yet $\mathsf{BQP} = \mathsf{QCMA} = \mathsf{QMA}$. In other words, as far as relativizing techniques are concerned, the classical and quantum versions of the P vs. NP question are completely uncoupled from one another.

Theorem 3 also represents progress toward a proof of the following conjecture, which might be the most alluring open problem that we leave.

▶ **Conjecture 5.** *There exists an oracle relative to which* $\mathsf{NP} \subseteq \mathsf{BQP}$ *but* $\mathsf{PH} \not\subset \mathsf{BQP}$*. Indeed, for every* $k \in \mathbb{N}$*, there exists an oracle relative to which* $\Sigma_k^{\mathsf{P}} \subseteq \mathsf{BQP}$ *but* $\Sigma_{k+1}^{\mathsf{P}} \not\subset \mathsf{BQP}$.

Conjecture 5 would provide spectacularly fine control over the relationship between BQP and PH, going far beyond Raz-Tal to show how BQP could, e.g., swallow the first 18 levels of PH without swallowing the 19th. To see the connection between Theorem 3 and Conjecture 5, suppose $\mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{BQP}^{\mathsf{NP}}$, and suppose also that $\mathsf{NP} \subseteq \mathsf{BQP}$. Then, as observed by Fortnow [20], this would imply

$$\mathsf{NP}^{\mathsf{NP}} \subseteq \mathsf{NP}^{\mathsf{BQP}} \subseteq \mathsf{BQP}^{\mathsf{NP}} \subseteq \mathsf{BQP}^{\mathsf{BQP}} = \mathsf{BQP},$$

(and so on, for all higher levels of PH), so that $\mathsf{PH} \subseteq \mathsf{BQP}$ as well. Hence, any oracle that witnesses Conjecture 5 also witnesses Theorem 3, so our proof of Theorem 3 is indeed a prerequisite to Conjecture 5.

At a high level, we prove Theorem 3 by showing that no $\mathsf{BQP}^{\mathsf{PH}}$ machine can solve the OR $\circ$ FORRELATION problem, in which one is given a long list of FORRELATION instances, and is tasked with distinguishing whether (1) all of the instances are uniformly random, or (2) at least one of the instances is Forrelated. A first intuition is that PH machines should gain no useful information from the input, just because FORRELATION "looks random" (by Raz-Tal), and hence a $\mathsf{BQP}^{\mathsf{PH}}$ machine should have roughly the same power as a BQP machine at deciding $\mathrm{OR} \circ \textsc{Forrelation}$. If one could show this, then completing the theorem would amount to showing that $\mathrm{OR} \circ \textsc{Forrelation}$ is hard for BQP machines, which easily follows from the BBBV Theorem [14].

Alas, initial attempts to formalize this intuition fail for a single, crucial reason: the possibility of homomorphic encryption! The Raz-Tal Theorem merely proves that FORRELATION is a strong form of encryption against PH algorithms. But to rule out a $\mathsf{BQP}^{\mathsf{PH}}$ algorithm for $\mathrm{OR} \circ \textsc{Forrelation}$, we *also* have to show that one cannot take a collection of FORRELATION instances and transform them, by means computable in PH, into a single FORRELATION instance whose solution is the OR of the solutions to the input instances. Put another way, we must show that AC^0 circuits of constant depth and $2^{\mathrm{poly}(n)}$ size cannot homomorphically evaluate the OR function, when the encryption is done via the FORRELATION problem.

More generally, we even have to show that AC^0 circuits cannot transform the "ciphertext" into *any* string that could later be decoded by an efficient quantum algorithm. Theorem 3 accomplishes this with the help of an additional structural property of AC^0 circuits: our concentration theorem for block sensitivity. Loosely speaking, the concentration theorem implies that, with overwhelming probability, any small AC^0 circuit is insensitive to toggling between a yes-instance and a neighboring no-instance of the $\mathrm{OR} \circ \textsc{Forrelation}$ problem. This, together with the BBBV Theorem [14], then implies that such "homomorphic encryption" is impossible.

We also achieve the following converse to Theorem 3:

▶ **Theorem 6.** *There exists an oracle relative to which* $\mathsf{BQP}^{\mathsf{NP}} \not\subset \mathsf{PH}^{\mathsf{BQP}}$*, and even* $\mathsf{BQP}^{\mathsf{NP}} \not\subset \mathsf{PH}^{\mathsf{PromiseBQP}}$*.*

Note that an oracle relative to which $\mathsf{BQP}^{\mathsf{NP}} \not\subset \mathsf{NP}^{\mathsf{BQP}}$ is almost trivial to achieve, for example by considering a problem in coNP. However, $\mathsf{BQP}^{\mathsf{NP}} \not\subset \mathsf{PH}^{\mathsf{BQP}}$ is much harder. At a high level, rather than considering the composed problem $\mathrm{OR} \circ \textsc{Forrelation}$, we now need to consider the reverse composition: $\textsc{Forrelation} \circ \mathrm{OR}$, a problem that's clearly in $\mathsf{BQP}^{\mathsf{NP}}$, but plausibly not in $\mathsf{PH}^{\mathsf{BQP}}$. The key step is to show that, when solving $\textsc{Forrelation} \circ \mathrm{OR}$, *any* $\mathsf{PH}^{\mathsf{BQP}}$ *machine can be simulated by a* PH *machine*: the BQP oracle is completely superfluous! Once we've shown that, $\textsc{Forrelation} \circ \mathrm{OR} \notin \mathsf{PH}$ then follows immediately from Raz-Tal.

For our next result, recall that QMA, or *Quantum Merlin-Arthur*, is the class of problems for which a yes-answer can be witnessed by a polynomial-size quantum state. Perhaps our second most interesting result is this:

▶ **Theorem 7.** PP *is not contained in the* "QMA hierarchy"*, consisting of constant-depth towers of the form* $\mathsf{QMA}^{\mathsf{QMA}^{\mathsf{QMA}^{\cdots}}}$*, with probability* 1 *relative to a random oracle.*[2]

Note that $\mathsf{PP} = \mathsf{PostBQP}$, where $\mathsf{PostBQP}$ denotes BQP augmented with the power of postselection [1], and so Theorem 7 contrasts with the classical containment $\mathsf{PostBPP} \subseteq \mathsf{BPP}^{\mathsf{NP}} \subseteq \mathsf{PH}$ [26, 33]. Nevertheless, before this paper, to our knowledge, it was not even

[2] Actually, our formal definition of the QMA hierarchy is more general than the version given here, in order to accommodate recursive queries to QMA promise problems. This only makes our separation stronger.

known how to construct an oracle relative to which $\mathsf{PP} \not\subset \mathsf{BQP}^{\mathsf{NP}}$, let alone classes like $\mathsf{BQP}^{\mathsf{NP}^{\mathsf{BQP}^{\mathsf{NP}^{\cdots}}}}$ or $\mathsf{QCMA}^{\mathsf{QCMA}^{\mathsf{QCMA}^{\cdots}}}$, which are contained in the QMA hierarchy. The closest result we are aware of is due to Kretschmer [32], who gave a *quantum* oracle relative to which $\mathsf{BQP} = \mathsf{QMA} \neq \mathsf{PostBQP}$.

Perhaps shockingly, our proof of Theorem 7 can be extended even to show that PP is not in, say, $\mathsf{QMIP}^{\mathsf{QMIP}^{\mathsf{QMIP}^{\cdots}}}$ relative to a random oracle, where QMIP means Quantum Multi-prover Interactive Proofs with entangled provers. This is despite the breakthrough results of Reichardt, Unger, and Vazirani [39], and more recently Ji, Natarajan, Vidick, Wright, and Yuen [30], which showed that in the *unrelativized* world, $\mathsf{QMIP} = \mathsf{MIP}^* = \mathsf{RE}$ (where MIP^* means QMIP with classical communication only, and RE means Recursively Enumerable), so in particular, QMIP contains the halting problem. This underscores the dramatic extent to which results like $\mathsf{QMIP} = \mathsf{RE}$ are nonrelativizing!

Theorem 7 can also be understood as showing that in the black-box setting, there is no quantum analogue of Stockmeyer's approximate counting algorithm [46]. For a probabilistic algorithm M that runs in $\mathrm{poly}(n)$ time and an error bound $\varepsilon \geq \frac{1}{\mathrm{poly}(n)}$, the approximate counting problem is to estimate the acceptance probability of M up to a multiplicative factor of $1 + \varepsilon$. Stockmeyer's algorithm [46] gives a relativizing $\mathrm{poly}(n)$-time reduction from the approximate counting problem to a problem in the third level of the polynomial hierarchy, and crucially relies on pulling the randomness out of M. In structural complexity terms, Stockmeyer's algorithm can be reinterpreted as showing that $\mathsf{SBP} \subseteq \mathsf{PH}$ relative to all oracles, where SBP is the complexity class defined in [16] that captures approximate counting.

One might wonder: is there a version of Stockmeyer's algorithm for the *quantum* approximate counting problem, where we instead wish to approximate the acceptance probability of a quantum algorithm? In particular, is SBQP, the complexity class that captures quantum approximate counting [33], contained in the QMA hierarchy?[3] Kuperberg [33] showed that $\mathsf{PP} \subseteq \mathsf{P}^{\mathsf{SBQP}}$, so it follows that $\mathsf{PP} \subseteq \mathsf{QMAH}$ if and only if $\mathsf{SBQP} \subseteq \mathsf{QMAH}$, where QMAH denotes the QMA hierarchy. Thus, Theorem 7 implies that $\mathsf{SBQP} \not\subset \mathsf{QMAH}$ relative to a random oracle, implying that such a quantum analogue of Stockmeyer's algorithm does not exist in the black-box setting.[4] This demonstrates yet another case where a classical complexity result that relies on fixing randomness cannot be generalized to the quantum setting.

Notably, our proof of Theorem 7 does not appeal to Raz-Tal at all, but instead relies on a new random restriction lemma for the acceptance probabilities of quantum query algorithms. Our random restriction lemma shows that if one randomly fixes most of the inputs to a quantum query algorithm, then the algorithm's behavior on the unrestricted inputs can be approximated by a "simple" function (say, a small decision tree or small DNF formula). We then use this random restriction lemma to generalize the usual random restriction proof that, for example, $\textsc{Parity} \notin \mathsf{AC}^0$ [27].

Here is another noteworthy result that we are able to obtain, by combining random restriction arguments with lower bounds on quantum query complexity:

[3] We thank Patrick Rall (personal communication) for bringing this question to our attention.

[4] Note that this is just one of many possible ways that we could ask whether there exists a quantum analogue of Stockmeyer's algorithm. For example, one might consider alternative definitions of the quantum approximate counting task, such as the problem defined in [18] of approximating the number of witness states accepted by a QMA verifier. One might also consider other definitions of the "quantum polynomial hierarchy," some of which are explored in [23].

▶ **Theorem 8.** *For every $k \in \mathbb{N}$, $\Sigma_{k+1}^{\mathsf{P}} \not\subset \mathsf{BQP}^{\Sigma_k^{\mathsf{P}}}$ with probability 1 relative to a random oracle.*

Theorem 8 extends the breakthrough of Håstad, Rossman, Servedio, and Tan [28], who (solving an open problem from the 1980s) showed that PH is infinite relative to a random oracle with probability 1. Our result shows, not only that a random oracle creates a gap between every two successive levels of PH, but that quantum computing fails to bridge that gap.

Again, Theorem 8 represents a necessary step toward a proof of Conjecture 5, because if we had $\Sigma_{k+1}^{\mathsf{P}} \subseteq \mathsf{BQP}^{\Sigma_k^{\mathsf{P}}}$, then clearly $\Sigma_k^{\mathsf{P}} \subseteq \mathsf{BQP}$ would imply $\Sigma_{k+1}^{\mathsf{P}} \subseteq \mathsf{BQP}^{\mathsf{BQP}} = \mathsf{BQP}$.

Our last two theorems return to the theme of the autonomy of BQP.

▶ **Theorem 9.** *There exists an oracle relative to which $\mathsf{NP} \subseteq \mathsf{BQP}$, and indeed $\mathsf{BQP} = \mathsf{P}^{\#\mathsf{P}}$, and yet PH is infinite.*

Theorem 9 resolves an open problem of Aaronson [2]. As a simple corollary, we also obtain an oracle relative to which $\mathsf{BQP} \not\subset \mathsf{NP/poly}$, resolving a question of Aaronson, Cojocaru, Gheorghiu, and Kashefi [7].

For three decades, one of the great questions of quantum computation has been whether it can solve NP-complete problems in polynomial time. Many experts guess that the answer is no, for similar reasons as they guess that $\mathsf{P} \neq \mathsf{NP}$ – say, the BBBV Theorem [14], combined with our failure to find any promising leads for evading that theorem's assumptions in the worst case. But the fact remains that we have no structural evidence connecting the $\mathsf{NP} \not\subset \mathsf{BQP}$ conjecture to any "pre-quantum" beliefs about complexity classes. No one has any idea how to show, for example, that if $\mathsf{NP} \subseteq \mathsf{BQP}$ then $\mathsf{P} = \mathsf{NP}$ as well, or anything even remotely in that direction.

Given the experience of classical complexity theory, it would be reasonable to hope for a theorem showing that, if $\mathsf{NP} \subseteq \mathsf{BQP}$, then PH collapses – analogous to the Karp-Lipton Theorem [31], that if $\mathsf{NP} \subset \mathsf{P/poly}$ then PH collapses, or the Boppana-Håstad-Zachos Theorem [17], that if $\mathsf{NP} \subseteq \mathsf{coAM}$ then PH collapses. No such result is known for $\mathsf{NP} \subseteq \mathsf{BQP}$, once again because of the difficulty that there is no known way to pull the randomness out of a BQP algorithm. Theorem 9 helps to explain this situation, by showing that any proof of such a conditional collapse would have to be nonrelativizing. The proof of Theorem 9 builds, again, on the Raz-Tal Theorem. And this is easily seen to be necessary, since as we pointed out earlier, if $\mathsf{BQP} \subseteq \mathsf{AM}$, then $\mathsf{NP} \subseteq \mathsf{BQP}$ really *would* imply a collapse of PH.

▶ **Theorem 10.** *There exists an oracle relative to which $\mathsf{P} = \mathsf{NP} \neq \mathsf{BQP} = \mathsf{P}^{\#\mathsf{P}}$.*

Theorem 10 says, in effect, that there is no relativizing obstruction to BQP being inordinately powerful even while NP is inordinately weak. It substantially extends the Raz-Tal Theorem, that there is an oracle relative to which $\mathsf{BQP} \not\subset \mathsf{PH}$, to show that in some oracle worlds, BQP doesn't go just *slightly* beyond the power of PH (which, if $\mathsf{P} = \mathsf{NP}$, is simply the power of P), but *vastly* beyond it. Once again, this illustrates the difference between randomness and quantumness, because if $\mathsf{P} = \mathsf{NP}$, then $\mathsf{P} = \mathsf{BPP}$ for relativizing reasons.

We conjecture that Theorem 10 could be extended yet further, to give an oracle relative to which $\mathsf{P} = \mathsf{NP}$ and yet $\mathsf{BQP} = \mathsf{EXP}$, but we leave that problem to future work.

1.4 Proof Techniques

We now give rough sketches of the important ideas needed to prove our results. Here, in contrast to Section 1.3, we present the results in the order that they appear in the main text, which is roughly in order of increasing technical difficulty.

Our proofs of Theorem 9 and Theorem 10 serve as useful warm-ups, giving a flavor for how we use the Raz-Tal Theorem and oracle construction techniques in later proofs. In Theorem 9, to construct an oracle where $\mathsf{BQP} = \mathsf{P}^{\#\mathsf{P}}$ but PH is infinite, we start by taking a random oracle, which by the work of Håstad, Rossman, Servedio, and Tan [28, 41] is known to make PH infinite. Then, for each $\mathsf{P}^{\#\mathsf{P}}$ machine M, we add to the oracle an instance of the FORRELATION problem that encodes the behavior of M: if M accepts, we choose a Forrelated instance, while if M rejects, we choose a uniformly random instance. This gives a BQP machine the power to decide any $\mathsf{P}^{\#\mathsf{P}}$ language.[5]

It remains to argue that adding these FORRELATION instances does not collapse PH. We want to show that relative to our oracle, for every k, there exists a language in $\Sigma_{k+1}^{\mathsf{P}}$ that is not in Σ_k^{P}. This is where we leverage the Raz-Tal Theorem: because the FORRELATION instances look random to PH, we can show, by a hybrid argument, that a Σ_k^{P} algorithm's probability of correctly deciding a target function in $\Sigma_{k+1}^{\mathsf{P}}$ is roughly unchanged if we replace the FORRELATION instances with uncorrelated, uniformly random bits. But auxiliary random bits cannot possibly improve the success probability, and so a simple appeal to [28] implies that the $\Sigma_{k+1}^{\mathsf{P}}$ language remains hard for Σ_k^{P}.

The proof of Theorem 10, giving an oracle where $\mathsf{P} = \mathsf{NP} \neq \mathsf{BQP} = \mathsf{P}^{\#\mathsf{P}}$, follows a similar recipe to the proof of Theorem 9. We start with a random oracle, which separates PH from $\mathsf{P}^{\#\mathsf{P}}$, and then we add a second region of the oracle that puts $\mathsf{P}^{\#\mathsf{P}}$ into BQP by encoding all $\mathsf{P}^{\#\mathsf{P}}$ queries in instances of the FORRELATION problem. Next, we add a third region of the oracle that answers all NP queries, which has the effect of collapsing PH to P. Finally, we again leverage the Raz-Tal Theorem to argue that the FORRELATION instances have no effect on the separation between PH and $\mathsf{P}^{\#\mathsf{P}}$, because the FORRELATION instances look random to PH algorithms.

We next prove Theorem 8, that $\Sigma_{k+1}^{\mathsf{P}} \not\subset \mathsf{BQP}^{\Sigma_k^{\mathsf{P}}}$ relative to a random oracle. Our proof builds heavily on the proof by [28] that $\Sigma_{k+1}^{\mathsf{P}} \not\subset \Sigma_k^{\mathsf{P}}$ relative to a random oracle. Indeed, our proof is virtually identical, except for a single additional step.

[28]'s proof involves showing that there exists a function SIPSER$_d$ that is computable by a small AC^0 circuit of depth d (which corresponds to a $\Sigma_{d-1}^{\mathsf{P}}$ algorithm), but such that any small AC^0 circuit of depth $d-1$ (which corresponds to a $\Sigma_{d-2}^{\mathsf{P}}$ algorithm) computes SIPSER$_d$ on at most a $\frac{1}{2} + o(1)$ fraction of random inputs. This proof uses random restrictions, or more accurately, a generalization of random restrictions called *random projections* by [28]. Roughly speaking, the proof constructs a distribution $\mathcal{R}$ over random projections with the following properties:

[5] The careful reader might wonder: if we can encode the answers to $\mathsf{P}^{\#\mathsf{P}}$ machines, then what is to stop us from encoding the answers to some arbitrarily powerful class, such as EXP or RE, into the FORRELATION instances? For a $\mathsf{P}^{\#\mathsf{P}}$ machine M, we exploit the fact that we can always choose FORRELATION instances on oracle strings that cannot be queried by M. For example, if M runs in time t, then we can encode M's output into strings of length t^c for some $c > 1$, which remain accessible to a BQP machine with a larger polynomial running time. By contrast, if we tried to do the same for an EXP machine (say), we run into the problem that the machine whose behavior we are trying to encode could query the very encoding we are making of its output, and thus our oracle would be circularly defined.

(i) Any small AC^0 circuit C of depth $d-1$ "simplifies" with high probability under a random projection drawn from $\mathcal{R}$, say, by collapsing to a low-depth decision tree.

(ii) The target $\textsc{Sipser}_d$ function "retains structure" with high probability under a random projection drawn from $\mathcal{R}$.

(iii) The structure retained in (ii) implies that the original unrestricted circuit C fails to compute the $\textsc{Sipser}_d$ function on a large fraction of inputs.

To prove Theorem 8, we generalize step (i) above from $\Sigma_{d-2}^{\mathsf{P}}$ algorithms to $\mathsf{BQP}^{\Sigma_{d-2}^{\mathsf{P}}}$ algorithms. That is, if we have a quantum algorithm that queries arbitrary depth-$(d-1)$ AC^0 functions of the input, then we show that this algorithm's acceptance probability also "simplifies" under a random projection from $\mathcal{R}$. We prove this by combining the BBBV Theorem [14] with [28]'s proof of step (i).

We next move on to the proof of Theorem 3, where we construct an oracle relative to which $\mathsf{NP}^{\mathsf{BQP}} \not\subset \mathsf{BQP}^{\mathsf{PH}}$. Recall that we prove Theorem 3 by showing that no $\mathsf{BQP}^{\mathsf{PH}}$ machine can solve the $\mathrm{OR} \circ \textsc{Forrelation}$ problem. To establish this, imagine that we fix a "no" instance x of the $\mathrm{OR} \circ \textsc{Forrelation}$ problem, meaning that x consists of a list of $\sim 2^n$ $\textsc{Forrelation}$ instances that are all uniformly random (i.e. non-Forrelated). We can turn x into an adjacent "yes" instance y by randomly choosing one of the $\textsc{Forrelation}$ instances of x and changing it to be Forrelated.

Our proof amounts to showing that with high probability over x, an AC^0 circuit of size $2^{\mathrm{poly}(n)}$ is unlikely (over y) to distinguish x from y. Then, applying the BBBV Theorem [14], we can show that for most choices of x, a $\mathsf{BQP}^{\mathsf{PH}}$ algorithm is unlikely to distinguish x from y, implying that it could not have solved the $\mathrm{OR} \circ \textsc{Forrelation}$ problem.

Next, we notice that it suffices to consider what happens when, instead of choosing y by randomly flipping one of the $\textsc{Forrelation}$ instances of x from uniformly random to Forrelated, we instead choose a string z by randomly resampling one if the instances of x from the uniform distribution. This is because, as a straightforward consequence of the Raz-Tal Theorem (Theorem 2), if f is an AC^0 circuit of size $2^{\mathrm{poly}(n)}$, then $|\Pr_y[f(x) \neq f(y)] - \Pr_z[f(x) \neq f(z)]| \leq 2^{-\Omega(n)}$.

Our key observation is that the quantity $\Pr_z[f(x) \neq f(z)]$ is proportional to a sort of "block sensitivity" of f on x. More precisely, it is proportional to an appropriate averaged notion of block sensitivity, where the average is taken over collections of blocks that respect the partition into separate $\textsc{Forrelation}$ instances. This is where our block sensitivity concentration theorem comes into play:

▶ **Theorem 11.** *Let $f : \{0,1\}^N \to \{0,1\}$ be an AC^0 circuit of size* $\mathrm{quasipoly}(N)$ *and depth $O(1)$, and let $B = \{B_1, B_2, \ldots, B_k\}$ be a collection of disjoint subsets of $[N]$. Then for any t,*

$$\Pr_{x \sim \{0,1\}^N}[\mathsf{bs}_B^x(f) \geq t] \leq 4N \cdot 2^{-\Omega\left(\frac{t}{\mathrm{polylog}(N)}\right)},$$

where $\mathsf{bs}_B^x(f)$ denotes the block sensitivity of f on x with respect to B.

Informally, Theorem 11 says that the probability that an AC^0 circuit has B-block sensitivity $t \gg \mathrm{polylog}(N)$ on a random input x decays exponentially in t. This generalizes the result of Linial, Mansour, and Nisan [35] that the *average* sensitivity of AC^0 circuits is at most $\mathrm{polylog}(N)$. It also generalizes a concentration theorem for the sensitivity of AC^0 circuits that appeared implicitly in the work of Gopalan, Servedio, Tal, and Wigderson [24], by taking B to be the partition into singletons.[6] In fact, we derive Theorem 11 as a simple corollary of such a sensitivity tail bound for AC^0. For completeness, we will also prove our own sensitivity tail bound, rather than appealing to [24]. Our sensitivity tail bound follows from an AC^0 random restriction lemma due to Rossman [40].

To prove Theorem 4, which gives an oracle relative to which $\mathsf{P} = \mathsf{NP}$ but $\mathsf{BQP} \neq \mathsf{QCMA}$, we use a similar technique to the proof of Theorem 10. We first take the oracle constructed in Theorem 3 that contains instances of the OR ∘ Forrelation problem. Next, we add a second region of the oracle that answers all NP queries. This collapses PH to P. Finally, we use Theorem 3 to argue that these NP queries do not enable a BQP machine to solve the OR ∘ Forrelation problem, which is in QCMA.

We now move on to the proof of Theorem 6, that there exists an oracle relative to which $\mathsf{BQP}^{\mathsf{NP}} \not\subset \mathsf{PH}^{\mathsf{BQP}}$. Recall that our strategy is to show that no $\mathsf{PH}^{\mathsf{BQP}}$ machine can solve the Forrelation ∘ OR problem. We prove this by showing that with high probability, a $\mathsf{PH}^{\mathsf{BQP}}$ machine on a random instance of the Forrelation ∘ OR problem can be simulated by a PH machine, from which a lower bound easily follows from the Raz-Tal Theorem. This simulation hinges on the following theorem, which seems very likely to be of independent interest:

▶ **Theorem 12.** *Consider a quantum algorithm Q that makes T queries to an $M \times N$ array of bits x, where each length-N row of x contains a single uniformly random 1 and 0s everywhere else. Then for any $\varepsilon \gg \frac{T}{\sqrt{N}}$ and $\delta > 0$, there exists a deterministic classical algorithm that makes $O\left(\frac{T^5}{\varepsilon^4}\log\frac{T}{\delta}\right)$ queries to x, and approximates Q's acceptance probability to within additive error ε on a $1-\delta$ fraction of such randomly chosen x's.*

Informally, Theorem 12 says that any fast enough quantum algorithm can be simulated by a deterministic classical algorithm, with at most a polynomial blowup in query complexity, on almost all sufficiently sparse oracles. The crucial point here is that the classical simulation still needs to work, even in most cases where the quantum algorithm is lucky enough to find many "1" bits. We prove Theorem 12 via a combination of tail bounds and the BBBV hybrid argument [14].

In the statement of Theorem 12, we do not know whether the exponent of 5 on T is tight, and suspect that it isn't. We only know that the exponent needs to be at least 2, because of Grover's algorithm [25].

We remark that Theorem 12 bears similarity to a well-known conjecture that involves simulation of quantum query algorithms by classical algorithms. A decade ago, motivated by the question of whether $\mathsf{P} = \mathsf{BQP}$ relative to a random oracle with probability 1, Aaronson and Ambainis [3] proposed the following conjecture:

▶ **Conjecture 13** ([3, Conjecture 1.5]; attributed to folklore)**.** *Consider a quantum algorithm Q that makes T queries to $x \in \{0,1\}^N$. Then for any $\varepsilon, \delta > 0$, there exists a deterministic classical algorithm that makes* $\operatorname{poly}\left(T, \frac{1}{\varepsilon}, \frac{1}{\delta}\right)$ *queries to x, and approximates Q's acceptance probability to within additive error ε on a $1-\delta$ fraction of uniformly randomly inputs x.*

[6] Interestingly, [24]'s goal, in proving their concentration theorem for the sensitivity of AC^0, was to make progress toward a proof of the famous *Sensitivity Conjecture* – a goal that Huang [29] achieved shortly afterward using completely different methods. One happy corollary of this work is that, nevertheless, [24]'s attempt on the problem was not entirely in vain.

While Conjecture 13 has become influential in Fourier analysis of Boolean functions,[7] it remains open to this day. Theorem 12 could be seen as *the analogue of Conjecture 13 for sparse oracles* – an analogue that, because of the sparseness, turns out to be much easier to prove.

We conclude with the proof of Theorem 7, showing that PP is not contained in the QMA hierarchy relative to a random oracle. This is arguably the most technically involved part of this work. Recall that our key contribution, and the most important step of our proof, is a random restriction lemma for quantum query algorithms. In fact, we even prove a random restriction lemma for functions with low *quantum Merlin-Arthur* (QMA) *query complexity*: that is, functions f where a verifier, given an arbitrarily long "witness state," can become convinced that $f(x) = 1$ by making few queries to x. Notably, our definition of QMA query complexity does not care about the length of the witness, but only on the number of queries made by the verifier. This property allows us to extend our results to complexity classes beyond QMA, such as QMIP.

An informal statement of our random restriction lemma is given below:

▶ **Theorem 14.** *Consider a partial function* $f : \{0,1\}^N \to \{0,1,\perp\}$ *with* QMA *query complexity at most* $\mathrm{polylog}(N)$*. For some* $p = \frac{1}{\sqrt{N}\mathrm{polylog}(N)}$*, let* ρ *be a random restriction that leaves each variable unrestricted with probability* p*. Then* f_ρ *is* $\frac{1}{\mathrm{quasipoly}(N)}$*-close, in expectation over* ρ*, to a* $\mathrm{polylog}(N)$*-width DNF formula.*[8]

An unusual feature of Theorem 14 is that we can only show that f_ρ is *close* to a simple function *in expectation*. By contrast, Håstad's switching lemma for DNF formulas [27] shows that the restricted function reduces to a simple function *with high probability*, so in some sense our result is weaker. Additionally, unlike the switching lemma, our result has a quantitative dependence on the number of inputs N. Whether this dependence can be removed (so that the bound depends only on the number of queries) remains an interesting problem for future work.

With Theorem 14 in hand, proving that $\mathsf{PP} \not\subset \mathsf{QMA}^{\mathsf{QMA}^{\mathsf{QMA}^{\cdots}}}$ relative to a random oracle is conceptually analogous to the proof that $\mathsf{PP} \not\subset \mathsf{PH}$ relative to a random oracle [27]. We first view a $\mathsf{QMA}^{\mathsf{QMA}^{\mathsf{QMA}^{\cdots}}}$ machine as a small constant-depth circuit in which the gates are functions of low QMA query complexity. Then we want to argue that the probability that such a circuit agrees with the Parity function on a random input is small. We accomplish this via repeated application of Theorem 14, interleaved with Håstad's switching lemma for DNF formulas [27].

To elaborate further, we first take a random restriction that, by Theorem 14, turns all of the bottom-layer QMA gates into DNF formulas. Next, we apply another random restriction and appeal to the switching lemma to argue that these DNFs reduce to functions of low decision tree complexity, which can be absorbed into the next layer of QMA gates. Finally, we repeat as many times as needed until the entire circuit collapses to a low-depth decision tree. Since the Parity function reduces to another Parity function under any random restriction,

[7] In the context of Fourier analysis, the Aaronson-Ambainis Conjecture usually refers to a closely-related conjecture about influences of bounded low-degree polynomials; see e.g. [36, 37]. Aaronson and Ambainis [3] showed that this related conjecture implies Conjecture 13.

[8] By saying that f_ρ is "close" to a DNF formula, we mean that there exists a DNF g depending on ρ such that the fraction of inputs on which f_ρ and g agree is $1 - \frac{1}{\mathrm{quasipoly}(N)}$, in expectation over ρ. In the full version [8], we introduce some additional notation and terminology that makes it easier to manipulate such expressions, but we will not use them in this exposition.

we conclude that this decision tree will disagree with the reduced Parity function on a large fraction of inputs, and hence the original circuit must have disagreed with the Parity function on a large fraction of inputs as well.

Of course, the actual proof of Theorem 7 is more complicated because of the accounting needed to bound the error introduced from Theorem 14, but all of the important concepts are captured above.

We end with a few remarks on the proof ideas needed for Theorem 14. Essentially, the first step involves proving that if we take a function f computed by a quantum query algorithm Q, a random restriction ρ, and a uniformly random input x to f_ρ, then x likely contains a small set K of "influential" variables. These influential variables have the property that for any string y that agrees with x on K, $|\Pr[Q(x) = 1] - \Pr[Q(y) = 1]|$ is bounded by a small constant. Hence, K serves as a certificate for f_ρ's behavior on x.

Proving that such a K usually exists amounts to a careful application of the BBBV Theorem [14]. Finally, we generalize from quantum query algorithms to arbitrary QMA query algorithms by observing that we only need to keep track of the certificates for inputs x such that $f_\rho(x) = 1$. The DNF we obtain in Theorem 14 is then simply the OR of all of these small 1-certificates.

Due to space constraints, we defer to the full version of our paper [8] for complete proofs and additional discussion.

References

1 Scott Aaronson. Quantum computing, postselection, and probabilistic polynomial-time. *Proceedings of the Royal Society A*, 461:3473–3482, 2005. doi:10.1098/rspa.2005.1546.

2 Scott Aaronson. BQP and the polynomial hierarchy. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, STOC '10, pages 141–150, New York, NY, USA, 2010. Association for Computing Machinery. doi:10.1145/1806689.1806711.

3 Scott Aaronson and Andris Ambainis. The need for structure in quantum speedups. *Theory of Computing*, 10(6):133–166, 2014. doi:10.4086/toc.2014.v010a006.

4 Scott Aaronson and Andris Ambainis. Forrelation: A problem that optimally separates quantum from classical computing. *SIAM Journal on Computing*, 47(3):982–1038, 2018. doi:10.1137/15M1050902.

5 Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. *Theory of Computing*, 9(4):143–252, 2013. doi:10.4086/toc.2013.v009a004.

6 Scott Aaronson and Lijie Chen. Complexity-Theoretic Foundations of Quantum Supremacy Experiments. In Ryan O'Donnell, editor, *32nd Computational Complexity Conference (CCC 2017)*, volume 79 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:67, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.CCC.2017.22.

7 Scott Aaronson, Alexandru Cojocaru, Alexandru Gheorghiu, and Elham Kashefi. Complexity-Theoretic Limitations on Blind Delegated Quantum Computation. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 6:1–6:13, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.ICALP.2019.6.

8 Scott Aaronson, DeVon Ingram, and William Kretschmer. The acrobatics of BQP, 2021. arXiv:2111.10409.

9 Leonard Adleman. Two theorems on random polynomial time. In *19th Annual Symposium on Foundations of Computer Science (SFCS 1978)*, pages 75–83, 1978. doi:10.1109/SFCS.1978.37.

10 Leonard M. Adleman, Jonathan DeMarrais, and Ming-Deh A. Huang. Quantum computability. *SIAM Journal on Computing*, 26(5):1524–1540, 1997. doi:10.1137/S0097539795293639.

11 Frank Arute, Kunal Arya, Ryan Babbush, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. doi:10.1038/s41586-019-1666-5.

12 László Babai, Lance Fortnow, and Carsten Lund. Non-deterministic exponential time has two-prover interactive protocols. *computational complexity*, 1(1):3–40, 1991. doi:10.1007/BF01200056.

13 Theodore Baker, John Gill, and Robert Solovay. Relativizations of the P=?NP question. *SIAM Journal on Computing*, 4(4):431–442, 1975. doi:10.1137/0204037.

14 Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997. doi:10.1137/S0097539796300933.

15 Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997. doi:10.1137/S0097539796300921.

16 Elmar Böhler, Christian Glaßer, and Daniel Meister. Error-bounded probabilistic computations between MA and AM. *Journal of Computer and System Sciences*, 72(6):1043–1076, 2006. doi:10.1016/j.jcss.2006.05.001.

17 Ravi B. Boppana, Johan Håstad, and Stathis Zachos. Does co-NP have short interactive proofs? *Inf. Process. Lett.*, 25(2):127–132, May 1987. doi:10.1016/0020-0190(87)90232-8.

18 Sergey Bravyi, Anirban Chowdhury, David Gosset, and Pawel Wocjan. On the complexity of quantum partition functions, 2021. arXiv:2110.15466.

19 Michael J. Bremner, Richard Jozsa, and Dan J. Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A*, 467:459–472, 2010. doi:10.1098/rspa.2010.0301.

20 Lance Fortnow. Pulling out the quantumness [online]. December 2005. URL: https://blog.computationalcomplexity.org/2005/12/pulling-out-quantumness.html.

21 Lance Fortnow and John Rogers. Complexity limitations on quantum computation. In *Proceedings. Thirteenth Annual IEEE Conference on Computational Complexity (Formerly: Structure in Complexity Theory Conference) (Cat. No.98CB36247)*, pages 202–209, 1998. doi:10.1109/CCC.1998.694606.

22 Merrick Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984. doi:10.1007/BF01744431.

23 Sevag Gharibian, Miklos Santha, Jamie Sikora, Aarthi Sundaram, and Justin Yirka. Quantum Generalizations of the Polynomial Hierarchy with Applications to QMA(2). In Igor Potapov, Paul Spirakis, and James Worrell, editors, *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*, volume 117 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 58:1–58:16, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.MFCS.2018.58.

24 Parikshit Gopalan, Rocco Servedio, Avishay Tal, and Avi Wigderson. Degree and sensitivity: tails of two distributions, 2016. Earlier version in CCC 2016. arXiv:1604.07432.

25 Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, pages 212–219, New York, NY, USA, 1996. Association for Computing Machinery. doi:10.1145/237814.237866.

26 Yenjo Han, Lane A. Hemaspaandra, and Thomas Thierauf. Threshold computation and cryptographic security. *SIAM Journal on Computing*, 26(1):59–78, 1997. doi:10.1137/S0097539792240467.

27 Johan Håstad. *Computational Limitations of Small-Depth Circuits.* MIT Press, Cambridge, MA, USA, 1987.

28 Johan Håstad, Benjamin Rossman, Rocco A. Servedio, and Li-Yang Tan. An average-case depth hierarchy theorem for Boolean circuits. *J. ACM*, 64(5), August 2017. doi:10.1145/3095799.

29 Hao Huang. Induced subgraphs of hypercubes and a proof of the sensitivity conjecture. *Annals of Mathematics*, 190(3):949–955, 2019. doi:10.4007/annals.2019.190.3.6.

30 Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. MIP*=RE, 2020. arXiv:2001.04383.

31 Richard M. Karp and Richard J. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing*, STOC '80, pages 302–309, New York, NY, USA, 1980. Association for Computing Machinery. doi:10.1145/800141.804678.

32 William Kretschmer. Quantum Pseudorandomness and Classical Complexity. In Min-Hsiu Hsieh, editor, *16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021)*, volume 197 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 2:1–2:20, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.TQC.2021.2.

33 Greg Kuperberg. How hard is it to approximate the Jones polynomial? *Theory of Computing*, 11(6):183–219, 2015. doi:10.4086/toc.2015.v011a006.

34 Clemens Lautemann. BPP and the polynomial hierarchy. *Information Processing Letters*, 17(4):215–217, 1983. doi:10.1016/0020-0190(83)90044-3.

35 Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, Fourier transform, and learnability. *J. ACM*, 40(3):607–620, July 1993. doi:10.1145/174130.174138.

36 Ashley Montanaro. Some applications of hypercontractive inequalities in quantum information theory. *Journal of Mathematical Physics*, 53(12):122206, 2012. doi:10.1063/1.4769269.

37 Ryan O'Donnell and Yu Zhao. Polynomial Bounds for Decoupling, with Applications. In Ran Raz, editor, *31st Conference on Computational Complexity (CCC 2016)*, volume 50 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 24:1–24:18, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.CCC.2016.24.

38 Ran Raz and Avishay Tal. Oracle separation of BQP and PH. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, pages 13–23, New York, NY, USA, 2019. Association for Computing Machinery. doi:10.1145/3313276.3316315.

39 Ben W. Reichardt, Falk Unger, and Umesh Vazirani. A classical leash for a quantum system: Command of quantum systems via rigidity of CHSH games. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, ITCS '13, pages 321–322, New York, NY, USA, 2013. Association for Computing Machinery. doi:10.1145/2422436.2422473.

40 Benjamin Rossman. An entropy proof of the switching lemma and tight bounds on the decision-tree size of AC0. Manuscript, 2017. URL: https://users.cs.duke.edu/~br148/logsize.pdf.

41 Benjamin Rossman, Rocco A. Servedio, and Li-Yang Tan. Complexity theory column 89: The polynomial hierarchy, random oracles, and Boolean circuits. *SIGACT News*, 46(4):50–68, December 2015. doi:10.1145/2852040.2852052.

42 Adi Shamir. IP = PSPACE. *J. ACM*, 39(4):869–877, October 1992. doi:10.1145/146585.146609.

43 Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Review*, 41(2):303–332, 1999. doi:10.1137/S0036144598347011.

44 Daniel R. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997. doi:10.1137/S0097539796298637.

45 Michael Sipser. A complexity theoretic approach to randomness. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, STOC '83, pages 330–335, New York, NY, USA, 1983. Association for Computing Machinery. doi:10.1145/800061.808762.

46 Larry Stockmeyer. The complexity of approximate counting. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, STOC '83, pages 118–126, New York, NY, USA, 1983. Association for Computing Machinery. doi:10.1145/800061.808740.

47 Seinosuke Toda. PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 20(5):865–877, 1991. doi:10.1137/0220053.

48 John Watrous. Succinct quantum proofs for properties of finite groups. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 537–546. IEEE, 2000. doi:10.1109/SFCS.2000.892005.

49 Han-Sen Zhong, Hui Wang, Yu-Hao Deng, Ming-Cheng Chen, Li-Chao Peng, Yi-Han Luo, Jian Qin, Dian Wu, Xing Ding, Yi Hu, Peng Hu, Xiao-Yan Yang, Wei-Jun Zhang, Hao Li, Yuxuan Li, Xiao Jiang, Lin Gan, Guangwen Yang, Lixing You, Zhen Wang, Li Li, Nai-Le Liu, Chao-Yang Lu, and Jian-Wei Pan. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020. doi:10.1126/science.abe8770.