On Certified Randomness from Quantum Advantage Experiments

Roozbeh Bassirian[*]
Thesis Advisor: Bill Fefferman

Abstract

Recently Aaronson proposed a certified randomness protocol based on random circuit sampling (RCS), which is the leading proposal for achieving quantum advantage. This protocol relies on a non-standard complexity-theoretic conjecture. In this paper, we unconditionally prove two versions of this conjecture in the black-box setting.

Contents

1 Introduction

Certified randomness refers to randomness expansion protocols, where a small (secret) truly random binary seed s can be extended to x where $|x| \gg |s|$ and it is also statistically close to uniformly random, by interacting with an *untrusted* device. Originally device-independent (private) certified randomness protocols were proposed by using non-signalling quantum devices that violate the Bell inequality. Informally, assuming we are given two non-signalling devices, [PAM$^+$10] lower bounds the min-entropy of their output distribution by a function of their CHSH correlation function:

$$I = \sum_{x,y} (-1)^{xy} \left[\mathbf{Pr}(a = b|xy) - \mathbf{Pr}(a \neq b|xy) \right]$$

[*]Department of Computer Science, University of Chicago. Email:roozbeh@uchicago.edu

And one can use the CHSH violation score to certify the min-entropy of the output distribution, even while conditioned on a quantum adversary that holds a system correlated with these devices. Lastly, we can use randomness extractors [GUV09], to extend a small random seed using this high min-entropy source.

Aaronson's certified randomness proposal uses random circuit sampling (RCS) to construct the high min-entropy distribution. In RCS, given a random quantum circuit, we are asked to sample from the output distribution of the circuit. Informally, since the output distribution of a random quantum circuit has high min-entropy, if we truly get samples from the output distribution, it is clear that we can use extractors to expand our random seed. However, verifying that one has done RCS faithfully is difficult and requires exponential samples from the device [VV17, HKEG19]. To reduce the sample complexity, current experiments only use polynomial number of samples in their statistical tests to verify RCS, i.e. Heavy Output Generation (HOG), Linear Cross Entropy (LXEB) [AC17]. Intuitively, these statistical tests check whether slightly *heavier* outcomes happen more often than *light* outcomes. Even if we assume that passing these statistical tests are classically intractable, it is not clear what passing HOG-like [1] tests imply about min-entropy of the output distribution. Even though doing RCS faithfully passes these tests, a quantum algorithm might pass these tests without performing RCS.

To get past this issue, Aaronson uses a custom complexity-theoretic conjecture called the "long list quantum supremacy verification" assumption (LLQSV). In this work we prove two versions of LLQSV in the black-box setting to give further evidence for his certified randomness protocol.

1.1 Motivation and Background

As mentioned in [PAM+10], one disadvantage of Bell inequality protocols is the *experimental* assumption. Not only it is not easy to construct these systems in labs, we need to control the devices to monitor any unwanted communication in the adversarial scenario, in which we do not trust the manufacturer of the device. This make these devices hard to use for "public" certified random number generation. This is not the case for Aaronson's protocol as long as we assume the device is implementing a quantum process.

Furthermore, the computational tasks performed in these quantum advantage tests are toy problems designed to cater to the strengths of the experimental device, and the next natural step after achieving "quantum advantage" is for NISQ devices to perform a *useful* computational task. Aaronson's protocol is based on quantum random circuit sampling (RCS), the same computational task which was implemented in the Google [AAB+19] and second USTC [ZCC+21] experiments.

We first start by restating the definition of quantum random circuits in RCS. The distribution over random circuits is defined over a fixed circuit *architecture*. Informally, an architecture $\mathcal{A}$ is an outline of a quantum circuit on n qubit. One can sample a random circuit by specifying each gate and drawing each gate Haar randomly (or from a finite set of local gates). Now consider the following problem:

Problem 1. *Given a random quantum circuit C on n qubits, output a sample s such that:*

$$\mathop{\mathbf{E}}_{C}\Big[P_C(s)\Big] \geq \frac{b}{N}$$

where $P_C(s) = |\langle s| C |0^n\rangle|^2$ and $b = 1+\varepsilon$ and $\varepsilon = poly(1/n)$.

[1] We note other statistical tests such as cross-entropy [BIS+18] and linear-cross-entropy [AAB+19, AG19] are very similar, and we refer to these collectively as "HOG-like tests".

The first step to get the protocol proposed by [Aar20] is to prove that any quantum algorithm that solves Problem 1 has linear min-entropy. Informally, this implies that the trivial quantum algorithm that samples from the output distribution of C is almost optimal. We already know that the trivial quantum algorithm that samples from the output distribution of C is a solution for this problem [PT56, BIS$^+$18], and in fact we have strong evidence that current NISQ devices are capable of solving this problem [AAB$^+$19]. It is easy to see that one can feed the output distribution of an algorithm solving Problem 1 to randomness extractors such as [GUV09] and extract $O(n)$ number of random bits. However, at this stage we, we require even more random bits to generate the starting random circuit.

Now suppose we have access to a machine that solves Problem 1. Then consider the following protocol:

1. Generate pseudorandom n qubit quantum circuits $C_1, C_2, \ldots, C_k$, where $k = \text{poly}(n)$.
2. Get samples $s_1, \ldots, s_T$, and verify that the samples have the promised property (in exponential time).
3. Use a classical randomness extractor to extract random bits from $(s_1, \ldots, s_k)$.

Notice that we can choose the random circuit from a sub-exponentially secure pseudorandom distribution (or even secure against QSZK [Aar18a]) and tweak the security parameter to boost the number of random bits we start from in a single round. For completeness, we summarize how certified random number generation is possible from "long list" conjectures. We already know that the trivial quantum algorithm that samples from the output distribution of C is a solution for this problem [PT56, BIS$^+$18], and in fact we have strong evidence that current NISQ devices are capable of solving this problem [AAB$^+$19]. If we can prove that any algorithm in BQP for Problem 1 has min-entropy larger than the random seed required by our pseudorandom generator, we can connect cryptographic assumptions to certified random bit generation. Our main tool for proving such a claim is the hardness of LLQSV. Suppose we are given access to oracle $\mathcal{O}$ which is a long (exponentially large) list of n qubit random quantum circuits $C_1, C_2, \ldots, C_T$ paired with strings $s_1, s_2, \ldots, s_T$. We can informally define the Long List Hardness Assumption (LLHA):

Conjecture 2. *Given oracle access to $\mathcal{O}$ it is QCAM-Hard to distinguish the following two cases:*

1. *Each s_i is sampled uniformly at random.*
2. *Each s_i is sampled from the output distribution of C_i.*

To summarize the overall proof strategy, we focus on refuting the existence of a deterministic BQP algorithm for Problem 1. The key idea is that if we have access to a deterministic function solving Problem 1, we can use this function in an approximate counting argument to count the number of collisions between the output of that function and the output of the quantum circuit. Furthermore, if the list in the conjecture is long enough, we can use Chernoff bound to separate the two cases which gives a QCAM protocol for LLQSV. Suppose we are given an instance of LLQSV, where $T = N^3 = 2^{3n}$. Let $S = \sum_i^T P_{C_i}(s_i)$. Then by Hoeffding's inequality we have:

$$\mathbf{Pr}\left[S \leq bN^2 - \frac{\varepsilon}{2}N^2\right] \geq \exp(-\frac{\varepsilon^2 N^4}{N^3}) \geq \exp(-O(n))$$

Let $V = |\{i | s_i \in Q(C_i)\}|$. Next we can use Chernoff bound to separate the two cases:

1. $\mathbf{Pr}\left[V \geq (1+\delta)N^2\right] \leq \exp(-\frac{\delta^2}{2+\delta}N^2) \leq \exp(-N)$

2. $\mathbf{Pr}\left[V \leq (1-\delta)((1+\frac{\varepsilon}{2})N^2)\right] \leq \exp(-\frac{\delta}{2}(1+\frac{\varepsilon}{2})N^2) \leq \exp(-N)$

By setting $\delta = \varepsilon^2$, with high probability:

1. In the uniform case $V < (1+\varepsilon^2)N^2$.
2. Otherwise $V > (1+\frac{\varepsilon}{4})N^2$

Lastly, we can use approximate counting to estimate this value up to a multiplicative factor of $(1+\varepsilon^2)$ to decide between these two cases with high probability. In [Aar20], Aaronson provides further evidence that this argument can be extended to constant min-entropy algorithms and provide methods to accumulate entropy.

1.2 Our Results

In his initial certified randomness proposal [Aar18a, Aar20], Aaronson gave a reduction from the problem of certifying min-entropy to a certain decision problem called the Long List Quantum Supremacy Verification (LLQSV) problem. The LLQSV problem essentially asks how hard it is to distinguish exponentially many samples from RCS from uniformly random numbers. Aaronson showed that if this problem lies outside of QCAM then no efficient quantum algorithm can pass the HOG test with low min-entropy. Analogously to [AC17], Aaronson explored potential algorithms for this problem and found they did not solve it, and subsequently conjectured such a QCAM algorithm does not exist. This is a custom assumption, and therefore deserves scrutiny – it is simultaneously conjecturing this problem lies outside of BQP, outside of AM, and beyond. In our second result, we give evidence towards its validity in the black-box setting. Namely, we show that the black-box LLQSV problem lies outside of BQP and even outside of PH (and hence outside of AM).

Theorem 3. *(informal) No BQP or PH algorithm can solve black-box LLQSV. Furthermore, our lower-bound extends even to the case in which the quantum algorithm is allowed to make $O(2^{n/10})$ queries, and the relation in the PH algorithm can be computed in time $O(2^{n/c})$ for some constant c that depends on the level of polynomial hierarchy.*

We give two independent proofs of the PH lower bound and one of the BQP lower-bound. The first PH bound and the BQP bound are based on extending lower bounds for the MAJORITY function. The second PH bound is based on introducing a black-box variant of LLQSV that we call SquaredForrelation, which is also closely related to the Forrelation problem. In this problem we are given black-box access to two functions $f, g : \{0,1\}^n \rightarrow \{\pm 1\}$, and are asked to distinguish whether f, g are chosen uniformly at random or whether g is correlated with the heavy Fourier coefficients of f. It is simple to show that the SquaredForrelation problem is strictly easier than LLQSV, because it reduces to LLQSV and it also admits a BQP algorithm. Intuitively this allows one to sample heavy elements of the Fourier spectrum and therefore recreate a long list of samples. Nevertheless, we prove that even the SquaredForrelation problem is not in PH. This simultaneously proves a lower bound on black-box LLQSV and gives another oracle separation of BQP vs PH [RT19].

1.3 Proof Outline

. First we formally define the Black-Box LLQSV problem:

Problem 4 (black-box LLQSV)**.** *Given oracle access to $\mathcal{O}$, distinguish the following two cases:*

- *$\mathcal{U}$: For each i, f_i and s_i are chosen uniformly at random.*
- *$\mathcal{D}$: For each i, f_i is a random Boolean function, and s_i is sampled according to $\widehat{f_i}^2$.*

We observe there is a close connection between black-box LLQSV and the MAJORITY problem. To see this we notice that in either case the probability of sampling a pair (f, s) only depends on the magnitude of the Fourier coefficient of f at s, or equivalently $|h(f \cdot \chi_s) - N/2|$, where h is the hamming weight and χ_s is the s-Fourier character (i.e., $\chi_s(x) = (-1)^{x \cdot s})$). This allows us to write both distributions as a linear combination of simpler distributions $\mathcal{U}_d^N$ – the uniform distribution over all strings of hamming weight $N/2 + d$ or $N/2 - d$.

An averaging argument then implies that any algorithm for black-box LLQSV should also distinguish a pair of distributions in the linear combinations. To prove the hardness of this task, a hybrid argument allows us to focus on distinguishability of $\mathcal{U}_0^N$ and $\mathcal{U}_d^N$, and since we are working with random Boolean functions, d is bounded by $\mathrm{poly}(n)/\sqrt{N}$. Lastly, we use known lower bounds for MAJORITY to prove that these two distributions are indistinguishable for both BQP and PH algorithms. We discuss this in more details in Section 2.

Perhaps more interestingly, we can show a reduction from a similar problem to Forrelation, that we call SquaredForrelation, to black-box LLQSV. This enables us to get an alternative lower bound for black-box LLQSV, by proving a PH lower bound for SquaredForrelation. Crucially, it is not hard to see that SquaredForrelation is in BQP, which makes it a strictly easier problem than black-box LLQSV, and hence our lower bound stronger. SquaredForrelation also gives an alternative oracle separation of BQP and PH, that has distinct advantages from previous similar results.

One advantage is that the quantum algorithm for SquaredForrelation only makes *classical* queries to g. On one hand, this property plays a critical role in the reduction between SquaredForrelation and black-box LLQSV. Moreover, this property allows us to consider a more realistic instantiation of these oracles (compared to [RT19]), as explained next.

Recall that in [RT19], an oracle separation $\mathsf{BQP}^O \not\subseteq \mathsf{PH}^O$ was obtained with respect to the Forrelation problem. The problem asks to distinguish whether two functions f and g (given as oracles) are either completely random and independent or whether the Fourier transform of f correlates with g. The Forrelation problem can be solved via a simple efficient quantum algorithm that queries both f and g in superposition, but cannot be solved by a PH machine. One drawback of this problem is that there's seemingly no efficient implementation of the oracles under which the problem remains non-trivial. In particular, if f is a random small circuit, then it seems any small circuit g doesn't correlate with f's Fourier transform. If one settles for having small circuits only for f but not g (or vice versa), then the problem remains non-trivial, but even in this scenario the problem seems far from practical as a quantum algorithm would need to query an exponentially large oracle g in superposition.

Our new problem SquaredForrelation, which is a variant on Forrelation, has the feature that the quantum algorithm solving it works as follows: (i) query f in superposition once, then (ii) measure the entire state, and finally (3) query g classically. The problem is defined as follows:

Problem 5 (simplified SquaredForrelation)**.** *Given oracle access to Boolean functions $f, g : \{0,1\}^n \to \{\pm 1\}$, distinguish the following two cases:*

1. *f and g are both chosen uniformly at random.*
2. *f is chosen uniformly at random and g is the indicator of the heavy Fourier coefficients of f.*

where by a heavy Fourier coefficient, we mean that the squared coefficient is larger than its expected value $\frac{1}{N}$. The main difference between SquaredForrelation and Forrelation is that g is correlated with *squared* Fourier coefficients rather than the sign of Fourier coefficients.

Indeed the quantum algorithm that queries f in superposition, measures in the Hadamard basis, and then queries g classically solves the problem. Notice that this algorithm is a combination of a quantum and classical algorithm. This means that in principle, if f has a small circuit, then the quantum part of the algorithm can implemented efficiently, followed by a much longer classical "post-mortem" part that queries g.

In particular, following [AC17, Section 7.4], this suggests choosing the oracle f, the only function that is queried in superposition, from a pseudorandom family of functions. We have not yet explored these directions in this manuscript, but we point out that one difference between our problem and the problems from [AC17, Section 7.4] (called Fourier Fishing and Fourier Sampling) is that ours is a decision problem whereas theirs are sampling problems.

Lower Bound for black-box LLQSV via $\mathsf{SquaredForrelation}$ We summarize the second PH lower bound for black-box LLQSV in two steps. First we prove a PH lower bound for $\mathsf{SquaredForrelation}$, and then we show a reduction from $\mathsf{SquaredForrelation}$ to black-box LLQSV.

The first step of the proof proceeds by carefully incorporating the *squared* Fourier coefficients in the analysis of Raz and Tal [RT19]. Recall that the analysis of [RT19] proceeds by looking at a multivariate Gaussian as a stopped Brownian motion. This allows them to write the expected value of a multilinear function over the multivariate Gaussian as a telescopic sum, and bound each term in the sum independently. Finally, they show that it is possible to convert this continuous distribution to a discrete distribution $\mathcal{D}$ while preserving the expected value by truncating each coordinate to the interval $[-1, 1]$ followed by randomized rounding.

A nice feature about Gaussians that was heavily exploited in the analysis of [RT19] is that the sum of t independent Gaussians is again a Gaussian. When dealing with squares of Gaussians this is no longer true. Nevertheless, if $Z \sim \mathcal{N}(0, \sigma^2)$ then one can still write $Z^2 - \mathbf{E}[Z^2]$ as the result of a martingale composed of many small steps. To present $Z^2 - \mathbf{E}[Z^2]$ as a martingale, first observe that if X (the "past") and Δ (the "next step") are two independent zero-mean Gaussians, then

$$(X + \Delta)^2 - \mathbf{E}[(X + \Delta)^2] = (X^2 - \mathbf{E}[X^2]) + \Delta^2 + 2\Delta \cdot X - \mathbf{E}[\Delta^2]$$

and thus $\mathbf{E}[\Delta^2 + 2\Delta \cdot X - E[\Delta^2] \mid X] = 0$. Since a Gaussian $Z \sim \mathcal{N}(0, \sigma^2)$ can be written as a sum of t independent Gaussians $Z_1, \ldots, Z_t \sim \mathcal{N}(0, \sigma^2/t)$, we see that $Z^2 - \mathbf{E}[Z^2]$ may be written as sum of t small steps, each of the form $Z_i^2 + 2Z_i \cdot Z^{<i} - \mathbf{E}[Z_i^2]$, with expected value 0 even conditioned on the past $Z^{<i} = Z_1 + \ldots + Z_{i-1}$. The rest follows from the analysis of [RT19] that crucially relies on presenting the final distribution as a sum of many small steps with expected value 0. As a consequence, we obtain a new oracle O so that $\mathsf{BQP}^O \not\subseteq \mathsf{PH}^O$.

In our next step, the reduction from $\mathsf{SquaredForrelation}$ to black-box LLQSV, we make use of the fact that g is not queried in superposition by the quantum algorithm. This observation follows by rejection sampling according to g; intuitively this allows one to sample heavy elements of the Fourier spectrum and therefore recreate a long list of samples. So far we have worked with one instance of the $\mathsf{SquaredForrelation}$ problem, however, we prove that our lower bound easily extend to the case that we have oracle access to a (exponentially) long list of samples in $\mathsf{SquaredForrelation}$. Consequently, this shows that the black-box LLQSV problem is not in PH.

In our previous discussion we have two oversimplifications which hide technical difficulties that we face in our argument. First, unlike in our simplified $\mathsf{SquaredForrelation}$ problem, the actual $\mathsf{SquaredForrelation}$ problem concerns a distribution over functions f and g that are obtained by taking multivariate Gaussians and randomized rounding them to have support over the Boolean cube. Second, rejection sampling according to g samples a uniformly random heavy squared Fourier coefficient of f, which is not the same as the Fourier transform distribution itself. We address these differences in detail in Section 3.4. Intuitively, to fix this issue recall that the goal of these long list hardness results is to generate certified random numbers. We show we can do that even when the quantum algorithm can sample heavy outcomes according to the distribution generated by rejection sampling (i.e., we can think of this as a new benchmark test much like HOG, but with respect to a new distribution). Accordingly, we show that the same quantum algorithm that samples according to the Fourier transform distribution also outputs heavy outcomes from the rejection sampling distribution on average. Proving this requires combining the analysis of the success probability of the quantum algorithm for SquaredForrelation with tail bounds on the number of "heavy" coefficients.

2 Black Box LLQSV

In the next two sections, we consider black-box variants of LLQSV. Here, we focus on black-box LLQSV, where instead of random circuits we have oracle access to random Boolean functions $f_1, \ldots, f_m$ and the samples $s_1, \ldots, s_m$ are taken either uniformly at random or according to the squared Fourier coefficients of the functions. In this setting, to provide evidence for QCAM hardness, we unconditionally prove lower bounds against BQP and PH (And hence, against AM [BHZ87]). Recall that black-box LLQSV asks to distinguish samples from distributions $\mathcal{D}$ (samples according to the Fourier coefficients) and $\mathcal{U}$ (uniformly random samples). First we notice an alternative way to sample from distribution $\mathcal{D}$ that makes our analysis easier. Since $\forall z \in \{0,1\}^n : \mathbf{Pr}_{\mathcal{D}}[s_i = z] = \frac{1}{N}$, we can sample from the joint distribution of a random function and a sample according to the Fourier distribution by first sampling a uniformly random outcome s_i, and then picking a random function f_i weighted according to $\widehat{f_i}(s_i)^2$. Both of our results in this section follow two observations. Let $\mathcal{U}_d^N$ be the uniform distribution over the following set:

$$\{S \in \{0,1\}^N | h(S) = N/2 - d \vee h(S) = N/2 + d\}$$

Where, $h(S)$ is the hamming weight of string S. First, for both distributions $\mathcal{U}$ and $\mathcal{D}$ the distribution over $f_i \cdot \chi_{s_i}$ is a linear combination of $\mathcal{U}_0^N$,$\mathcal{U}_1^N, \ldots, \mathcal{U}_{N/2}^N$. In the uniform case, this is clear since $f_i \cdot \chi_{s_i}$ is a uniformly random string. For $\mathcal{D}$, hamming weight of $f_i \cdot \chi_{s_i}$ specifies the Fourier coefficient $\widehat{f_i}(s_i)$, so all strings with hamming weight $N/2 + d$ or $N/2 - d$ occur with equal probability. So, we can write both probability distributions as:

$$\sum_{D:=(d_1,\ldots,d_m)\in[N/2+1]^m} p_D \mathcal{U}_{d_1} \times \ldots \times \mathcal{U}_{d_m}$$

For some set of coefficients where $\sum_D p_D = 1$. Let us call these distributions $\chi_{\mathcal{D}}$ and $\chi_{\mathcal{U}}$ respectively.

Next, we use the fact that in both distributions $\mathcal{U}$ and $\mathcal{D}$ the marginal distribution over the functions is uniform. Thus, with high probability the Fourier coefficients are bounded:

Theorem 6. *Given a random Boolean function $f : \{\pm 1\} \to \{0,1\}$, the probability that a squared Fourier coefficient of f is larger than $\frac{p(n)^2}{N}$ is at most $2\exp(\frac{-p(n)^2}{6\ln N})$.*

Proof. This follows from a Chernoff bound. Let $X = h(f \cdot X_s)$. For every s, this is a uniformly random string, so we can write:

$$\mathbf{Pr}\left[X \leq \left(1 - \frac{p(n)}{\sqrt{N}}\right) N/2\right] \leq \exp(-p(n)^2/4)$$

$$\mathbf{Pr}\left[X \geq \left(1 + \frac{p(n)}{\sqrt{N}}\right) N/2\right] \leq \exp(-p(n)^2/6)$$

Combining these two and a union bound gives us the claim. □

Note that in both $\mathcal{D}$ and $\mathcal{U}$, f_i is a random Boolean function. So in either case by a union bound we can prove that with probability at least $1 - 2\exp(\frac{-p(n)^2}{6 \ln N \cdot \ln m})$ none of these functions have a squared Fourier coefficient larger than $\frac{p(n)^2}{N}$. So as long as $\ln m$ is polynomial in n we can choose $p(n)$ in a way that this event happens with an exponentially small probability.

It is clear that distinguishing $\mathcal{D}$ and $\mathcal{U}$ is harder than distinguishing $\chi_{\mathcal{U}}$ and $\chi_{\mathcal{D}}$ since given f_i, s_i we can compute $f_i \cdot \chi_{s_i}$. The other direction is not as clear since given a long list of strings $S_1, \ldots, S_m$, we need $s_1, \ldots, s_m$ to recover $f_1, \ldots, f_m$. However, given a sample from $\chi_{\mathcal{D}}$ or $\chi_{\mathcal{U}}$, we can easily get a sample from $\mathcal{D}$ or $\mathcal{U}$ respectively, by simply choosing uniformly random $s_1, \ldots, s_m$. So, assuming we have access to a long list of random bits, any distinguisher for $\chi_{\mathcal{U}}$ and $\chi_{\mathcal{D}}$ also distinguishes $\mathcal{D}$ and $\mathcal{U}$. As we will see in analysis having access to an extra random oracle does not give any extra computation power to the BQP or PH algorithm. Combining Theorem 6 with this fact implies that any distinguisher for $\mathcal{U}$ and $\mathcal{D}$ must also distinguish the following two distributions with high probability:

- $\chi'_{\mathcal{U}} = \sum_{d_1 \ldots d_m \in [p(n)^2\sqrt{N}]} p_D \mathcal{U}^N_{d_1} \times \ldots \times \mathcal{U}^N_{d_m}$
- $\chi'_{\mathcal{D}} = \sum_{d_1 \ldots d_m \in [p(n)^2\sqrt{N}]} q_D \mathcal{U}^N_{d_1} \times \ldots \times \mathcal{U}^N_{d_m}$

Since $\chi_{\mathcal{U}}$ and $\chi_{\mathcal{D}}$ are statistically close to $\chi'_{\mathcal{U}}$ and $\chi'_{\mathcal{D}}$ respectively. Furthermore, since $\sum_D p_D \leq 1$ and $\sum_D q_D \leq 1$, by an averaging argument this distinguisher can also distinguish two distributions in the sum with a $1/\text{poly}(n)$ advantage. This reduces the problem to indistinguishability of $\mathcal{U}_{\mathcal{D}}$ and $U_{D'}$, where d_i and d'_i are at most $p(n)\sqrt{N}$. Thus, to complete the proof we need to show BQP and PH lower bounds for the following problem:

Problem 7 ($\mathsf{BalanceChecking}$)**.** *Let $0 \leq d_i \leq p(n) \cdot \sqrt{N}$. Given oracle access to a list of strings $S = S_1, \ldots, S_m$ distinguish the following two cases:*

1. *$\mathcal{U}_D$: S_i is drawn from $\mathcal{U}^N_{d_i}$.*
2. *$\mathcal{U}_0$: S_i is drawn from $\mathcal{U}^N_0$.*

Note that indistinguishability of $\mathcal{U}_D$ and $\mathcal{U}_0$ implies indistinguishability of $\mathcal{U}_D$ and $\mathcal{U}_{D'}$. We say algorithm A solves $\mathsf{BalanceChecking}$ with advantage δ, if:

$$\left| \Pr_{S \sim \mathcal{U}_D}[A \text{ accepts } S] - \Pr_{S \sim \mathcal{U}_0}[A \text{ accepts } S] \right| \geq \delta$$

In our proofs, we use a hybrid argument to get the BQP lower bound. Furthermore, this reduction enables us to use an argument similar to what is given in [FSUV13, Aar10] to prove a bound against $\mathsf{AC^0}$. Informally, since this simplified problem is random self reducible, combined with a circuit lower bound due to Håstad we can beat the hybrid argument, and extend the lower bound to an exponentially long list. Notice that black-box LLQSV is slightly different than $\mathsf{BalanceChecking}$, since it also has access to a random string for each S_i (which is the outcome s_i). To complete our proof, we need to show that having access to these extra random strings do not give us any extra computation power. This is straightforward from the hybrid argument for the BQP lower bound, and we can again use random self-reducibility of $\mathsf{BalanceChecking}$ to show the same claim for the PH argument.

2.1 BQP Lower Bound for black-box LLQSV

Theorem 8. *Any T query quantum algorithm for* $\mathsf{BalanceChecking}$ *has advantage at most $p(n) \cdot TN^{-1/8}$.*

Proof. This follows from a BBBV style argument. Suppose that S is drawn from $\mathcal{U}_{d_1}^N \times \ldots \times \mathcal{U}_{d_m}^N$. We can modify each S_i in at most $p(n) \cdot \sqrt{N}$ entries to make each S_i balanced. Let this new string be S'. Let P_i be the set of possible entries for each string S_i, and $P = \bigcup_i P_i$. Then given a quantum algorithm A, we can write the query magnitude of each P_i as:

$$\sum_{t=1}^{T} \sum_{\substack{w \\ x \in P_i}} |\alpha_{t,x,w}|^2 \leq T_i$$

and $\sum_i T_i = T$. So if we choose a random subset $\Delta = \bigcup_i \Delta_i$ for the modified entries of each S_i, we can write the expected value of the query magnitude as:

$$\mathop{\mathbf{E}}_{\Delta} \Big[\sum_{t=1}^{T} \sum_{i}^{m} \sum_{\substack{w \\ x \in \Delta_i}} |\alpha_{t,x,w}|^2 \Big] = \sum_{i}^{m} \mathop{\mathbf{E}}_{\Delta_i} \Big[\sum_{t=1}^{T} \sum_{\substack{w \\ x \in \Delta_i}} |\alpha_{t,x,w}|^2 \Big] \leq \sum_{i}^{m} \frac{T_i p(n) \sqrt{N}}{2|P_i|} \leq \frac{p(n)T}{\sqrt{N}}$$

And by a Markov's inequality and Cauchy-Schwartz we get that for $1 - N^{-1/4}$ fraction of S':

$$\delta(A(S), A(S')) \leq 4 \cdot p(n) \cdot TN^{-1/8}$$

However, this random modification also preserves the distribution in this case. To sample according to $\mathcal{U}_0$, we can first sample from $\mathcal{U}_d$ and then randomly flip the excess bits to make the string balanced. So the advantage of A is at most:

$$N^{-1/4} + 4 \cdot p(n) \cdot TN^{-1/8} \leq 8 \cdot p(n) \cdot TN^{-1/8}$$

□

Now we can formally define the BQP bound of Theorem 3:

Theorem 3 (BQP Bound)**.** *Any quantum algorithm for black-box* LLQSV *with advantage $\delta = 1/\mathrm{poly}(n)$ must make at least $\Omega(N^{1/10})$ queries to the oracle.*

Proof. Suppose there exists a $O(N^{1/10})$ quantum query algorithm for black-box LLQSV. First we notice that the same hybrid argument works without any modification if in $\mathsf{BalanceChecking}$ we have access to (S_i, s_i), where s_i is a uniformly random string. Now we can use s_i to recover f by computing $S_i \cdot \chi_{s_i}$. From Theorem 6, any distinguisher for black-box LLQSV must also solve $\mathsf{BalanceChecking}$ (with the additional access to s_i) with advantage at least $\delta' = \delta - 2\exp(\frac{-p(n)^2}{6 \ln N \cdot \ln m})$. We can choose an appropriate polynomial p so that $\frac{p(n)^2}{6 \ln N \cdot \ln m} \geq n$. Then for large enough n, this implies $\delta' \geq \delta/2 = 1/\mathrm{poly}(n)$. However, we know that:

$$\delta' \leq 8 \cdot p(n) \cdot cN^{1/10} \cdot N^{-1/8} = O(N^{-1/40})$$

Which gives us a contradiction. □

2.2 PH Lower Bound for black-box LLQSV

We start from a well known AC^0 lower bound [Hås87]:

Lemma 9. *Given an N bit string S, any depth d circuit that accepts all S such that $h(S) = N/2+1$ and rejects them if $h(S) = N/2$ has size:*

$$\exp\big(\Omega(N^{1/(d-1)}\big)$$

First we notice that the same bound holds for distinguishing $h(S) = N/2$ from $|h(S)-N/2| = 1$. The next step is to prove analog of Theorem 8 against PH.

Theorem 10. *Any depth d circuit that solves $\mathsf{BalanceChecking}$ with $1/\text{poly}(n)$ advantage has size:*

$$\exp\big(\Omega(N^{1/(2d+4)})\big)$$

Proof. Suppose there exists a circuit C of size s and depth d that solves $\mathsf{BalanceChecking}$ with advantage $\delta = 1/\text{poly}(n)$. Using C we can construct a circuit C' of size $s' = \text{poly}(s, N)$ and depth $d+3$ that solves the problem in Lemma 9 for length $\sqrt{N}/p(n)$. Given a $\sqrt{N}/p(n)$ bit string t, for each i, we copy the string d_i times and then pad it with equal number of zeros and ones to have length N. We then take a random permutation of this string and flip all bits with probability $1/2$ to get S_i. Note that if the original string is balanced $S_i \sim \mathcal{U}_0^N$, and otherwise $S_i \sim \mathcal{U}_D^N$. Then we know:

$$\Big|\mathbf{Pr}\big[A \text{ accepts } S|t \text{ balanced}\big] - \mathbf{Pr}\big[A \text{ accepts } S|t \text{ not balanced}\big]\Big| \geq \delta$$

We can repeat this $r = \text{poly}(1/\delta, N)$ times. By a Chernoff bound, we get a $1/\delta$ gap with arbitrary large probability $\exp\big(-\text{poly}(N)\big)$, which is detectable by an approximate majority circuit (see [Aar10, Vio07] for more detail). Let us call this new circuit C'. We can then fix the randomness in a way that the new circuit C' succeeds on every input simultaneously. This new circuit has depth $d+3$ and size $\text{poly}(s, N)$. So s must be at least:

$$\exp\big(\Omega(\sqrt{N}^{-1/(d+2)}\big) = \exp\big(\Omega(N^{1/(2d+4)}\big)$$

□

Corollary 11. *Any depth d circuit that solves black-box LLQSV with $1/\text{poly}(n)$ advantage has size:*

$$\exp\big(\Omega(N^{1/(2d+4)})\big)$$

Proof. Similar to the BQP bound, the first step is to prove the same AC^0 bound for $\mathsf{BalanceChecking}$ even when the algorithm has additional access to a long list of random outcomes $s_1, \ldots, s_m$. This immediately follows from the proof idea of Theorem 10, since we can choose uniformly random s_i in the circuit for each instance separately (without accessing the oracle, while preserving the distribution), and later fix them in the circuit. Furthermore, from Theorem 6, any $1/\text{poly}(n)$ distinguisher for LLFS would also work for $\mathsf{BalanceChecking}$ with the additional outcome oracle by choosing the polynomial p to be large enough. □

Corollary 11 and the standard conversion between PH and AC^0 result in the PH lower bound of Theorem 3 for LLFS.

3 The SquaredForrelation Problem and black-box LLQSV

In this section we first prove an alternative oracle separation between BQP, and then show a reduction from SquaredForrelation to black-box LLQSV. To do this we build on the previous work of Raz and Tal [RT19] to define a distribution $\mathcal{D}$ over pairs of Boolean functions (f, g) that is indistinguishable from uniform in the polynomial hierarchy, and yet a simple quantum algorithm is able to distinguish them. This hardness result does not immediately follow from previous oracle separations and requires careful modifications of Raz and Tal analysis to make them work with squared Fourier coefficients for both the PH lower bound and the quantum algorithm. Furthermore, we extend these results to the "long list" setting, and prove that having oracle access to an exponentially long list of samples does not make the problem any easier.

Before getting into the formal results, let us start with the simplified version of SquaredForrelation from Section 1.3. One can choose the threshold parameter to be the median of squared Fourier coefficients of a random Boolean function, so that almost half of the coefficients are indicated as heavy on average, i.e $\mathbf{Pr}_{x,f}[g(x) = 1] \approx 1/2$. Then we can easily find random heavy Fourier coefficients by finding a random x such that $g(x) = 1$. This allows us to reduce the long list variant of this problem to a problem similar to black-box LLQSV, since we can use "rejection sampling" on g to pass HOG. The distribution $\mathcal{D}$ that we define next is similar to simplified SquaredForrelation in the sense that we use randomized rounding on squared Fourier coefficients and we can prove that on average, heavy Fourier coefficients have a slightly higher chance of having $g(x) = 1$. We leverage this to reduce SquaredForrelation to a black-box variant of LLQSV.

Let us first summarize the separation result that we are going to use to give evidence for LLQSV. Similar to [RT19], to show an oracle separation of BQP and PH it suffices to find a distribution $\mathcal{D}$ that is pseudorandom for AC^0 circuits, but not for efficient quantum algorithms making few queries. First we start by defining distribution $\mathcal{D}$ using truncated multivariate Gaussians. Let $n \in \mathbb{N}$, $N = 2^n$. Let $\varepsilon = 1/(C \ln N)$ for a constant $C \geq 20$ to be chosen later. Define $\mathcal{G}$ to be a multivariate Gaussian distribution over $\mathbb{R}^N \times \mathbb{R}^N$ with mean 0 and covariance matrix:

$$\varepsilon \cdot \begin{pmatrix} I_N & H_N \\ H_N & I_N \end{pmatrix}$$

To take samples from $\mathcal{G}$, we can first sample $X = x_1, \ldots, x_N \sim \mathcal{N}(0, \varepsilon)$, and let $Y = H_N \cdot X$. Define $\mathcal{G}'$ to be the distribution over $Z = (X, Y^2 - \varepsilon)$. Let $\operatorname{trnc}(a) = \min(1, \max(-1, a))$. The distribution $\mathcal{D}$ over $\{\pm 1\}^{2N}$, first draws $Z \sim \mathcal{G}'$. Then for each $i \in [2N]$ draws $z'_i = 1$ with probability $\frac{1+\operatorname{trnc}(z_i)}{2}$ and $z'_i = -1$ with probability $\frac{1+\operatorname{trnc}(z_i)}{2}$. Now we can define the following promise problem:

Problem 12 (SquaredForrelation)**.** *Given oracle access to Boolean functions $f, g : \{0,1\}^n \to \{\pm 1\}$, distinguish whether they are sampled according to $\mathcal{D}$ or uniformly at random.*

Recall that we can write the multilinear expansion of a Boolean function $F : \mathbb{R}^{2N} \to \mathbb{R}$ as:

$$F(z) = \sum_{S \subseteq [2N]} \widehat{F}(S) \prod_{i \in S} z_i$$

Similar to the original proof, it is not hard to see that the randomized rounding step does not change the expected value of the outcome for multilinear functions:

$$\mathop{\mathbf{E}}_{z \sim \mathcal{G}'}[F(\operatorname{trnc}(z))] = \mathop{\mathbf{E}}_{z' \sim \mathcal{D}}[F(z')]$$

Hence we can ignore step 3 in our analysis. More importantly, because of the choice of ε, truncations happen with negligible probability, and in the event that they happen it is possible to bound the difference.

$$\mathop{\mathbf{E}}_{z\sim\mathcal{G}'}[F(\mathrm{trnc}(z))] \approx \mathop{\mathbf{E}}_{z\sim\mathcal{G}'}[F(z)]$$

The rest of the analysis is to prove indistinguishability of $(X, Y^2-\varepsilon)$ from the uniform distribution. More specifically, to use the tail bound from [Tal17] for the Fourier coefficients of bounded depth circuits to bound:

$$\left|\mathbf{E}[F(X,Y^2-\varepsilon)] - \mathbf{E}[F(\mathcal{U}_{2N})]\right| = \left|\mathbf{E}[F(X,Y^2-\varepsilon) - F(0,0)]\right|$$

The idea is to use the same telescopic sum as Raz and Tal for analysing the expected value of F and think of $\mathcal{G}'$ as a Brownian motion, but also considering the squared inputs and the ε difference. One important step is to also break the ε bias in the telescopic sum. Let $t \in \mathbb{N}$, and for $i \in [t]$, let $X^{(i)} = X^{(i-1)} + \Delta_X^{(i)}$ and $Y^{(i)} = H_N \cdot X^{(i)}$, where $\Delta_X^{(i)} \sim \mathcal{N}(0, \varepsilon/t)$. Now we can write:

$$F(X^{(t)}, (Y^{(t)})^2 - \varepsilon) - F(0,0) = \sum_{i=1}^{t} \left\{ F\left(X^{(i)}, (Y^{(i)})^2 - \tfrac{\varepsilon\cdot i}{t}\right) - F\left(X^{(i-1)}, (Y^{(i-1)})^2 - \tfrac{\varepsilon\cdot(i-1)}{t}\right) \right\}$$

Then we can use known bounds for each term in the sum and the triangle inequality to bound the sum. In the limit of $t \to \infty$, we are dealing with a stochastic integral so one can also prove this result using stochastic calculus. However, we decided not to present the proof using the language of stochastic calculus, and instead picked t to be a large enough polynomial in N (similar to [RT19]).

3.1 Truncated Gaussians

It is not hard to see that any multilinear function $F : \mathbb{R}^{2N} \to \mathbb{R}$ still has similar expectation under $\mathcal{D}$ and $\mathcal{G}'$, where:

$$F(z) = \sum_{S\subseteq[2N]} \widehat{F}(S) \cdot \prod_{i\in S} z_i$$

First we need to show that:

$$\mathop{\mathbf{E}}_{z'\sim\mathcal{D}}\left[F\left(z'\right)\right] = \mathop{\mathbf{E}}_{z\sim\mathcal{G}'}[F(\mathrm{trnc}(z))]$$

And since we can still prove that the truncations happen with negligible probability, we only need to adapt the rest of the proof to work with squared Fourier coefficients. The proof of the first part works without any modifications from the definition of distributions $\mathcal{D}$ and $\mathcal{G}'$:

$$\begin{aligned}\mathbf{E}\left[F\left(z'\right) \mid z\right] &= \mathbf{E}\left[\sum_{S\subseteq[2N]} \widehat{F}(S)\cdot\prod_{i\in S} z'_i \,\middle|\, z\right] = \sum_{S\subseteq[2N]} \widehat{F}(S)\cdot\prod_{i\in S}\mathbf{E}\left[z'_i \mid z\right] \\ &= \sum_{S\subseteq[2N]} \widehat{F}(S)\cdot\prod_{i\in S}\mathrm{trnc}\left(z_i\right) = F(\mathrm{trnc}(z)) \qquad (1)\end{aligned}$$

The next fact is stated as Claim 5.1 in [RT19], and is unaffected by our modification.

Fact 13. *Let $F : \mathbb{R}^{2N} \to \mathbb{R}$ be a multilinear function that maps $\{\pm1\}^{2N}$ to $[-1,1]$. Let $z = (z_1, \ldots, z_{2N}) \in \mathbb{R}^{2N}$. Then, $|F(z)| \leq \prod_{i=1}^{2N} \max\left(1, |z_i|\right)$.*

The next two theorems are equivalent to Claim 5.2 and Claim 5.3 of [RT19] respectively, where they are proven over distribution $\mathcal{G}$. One can verify that they also hold over distribution $\mathcal{G}'$. In fact, by choosing ε to be small enough, they even hold over $\mathcal{G}'^{\mathrm{poly}(N)}$, where we are given oracle access to an exponentially long list of samples from $\mathcal{G}'$. This is crucial for extending this PH bound to the long list problem. We provide their proofs in Appendix A for completeness.

Theorem 14. *For any constant $c \geq 2$ there exists a choice of a constant C in the definition of $\mathcal{G}'$ such that:* $\mathbf{E}_{(x,y)\sim\mathcal{G}'}\left[\prod_{i=1}^{N}\max(1,|x_i|)\cdot\prod_{i=1}^{N}\max(1,|y_i|)\cdot\mathbb{1}_{(x,y)\neq\mathrm{trnc}(x,y)}\right] \leq 4\cdot N^{-c}$.

Theorem 15. *For any constant $c \geq 2$ there exists a choice of a constant C in the definition of $\mathcal{G}'$ such that the following holds. Let $0 \leq p, p_0$ such that $p + p_0 \leq 1$. Let $F : \mathbb{R}^{2N} \to \mathbb{R}$ be a multilinear function that maps $\{\pm1\}^{2N}$ to $[-1,1]$. Let $z_0 \in [-p_0, p_0]^{2N}$. Then,*

$$\mathop{\mathbf{E}}_{z\sim\mathcal{G}'}\left[\left|F\left(\mathrm{trnc}\left(z_0 + p\cdot z\right)\right) - F\left(z_0 + p\cdot z\right)\right|\right] \leq 8\cdot N^{-c}$$

3.2 Quantum Algorithm for Distinguishing $\mathcal{D}$ and U_{2N}

Here, for completeness, we provide a quantum algorithm for distinguishing U_{2N} and $\mathcal{D}$. Note that this quantum algorithm does not extend to solve LLFS, and does not contradict LLQSV. As we will show, a problem similar to LLFS is harder than $\mathsf{SquaredForrelation}$. However, we use these calculations for our later results. Given to Boolean functions $f, g : \{\pm1\}^N$ we use the following algorithm A to distinguish between functions sampled from $\mathcal{D}$ and U_{2N}:

1. Apply Fourier transform on f and sample x from the distribution induced by $\widehat{f}$.
2. Accept if $g(x) = 1$, and reject otherwise.

We can see that the success probability of this algorithm is given by $\sum_{x,g(x)=1}\widehat{f}(x)^2 = \frac{1+\varphi(f,g)}{2}$, where $\varphi(f,g) = \varphi(X,Y)$ is defined as follows:

$$\varphi(X,Y) = \frac{1}{N}\sum_i(\sum_j H_{ij}X_i)^2 Y_i = \mathbf{Pr}[A \text{ Accepts}] - \mathbf{Pr}[A \text{ Rejects}]$$

We first analyze this quantity for when the samples are taken from $\mathcal{G}'$. In fact, we can show that the multilinear part of $\varphi(X,Y)$, denoted by $\varphi_{i\neq j}(X,Y)$ is large on average. Lastly we can use Theorem 15 to show that a similar bound holds for samples from $\mathcal{D}$ while the quadratic part of $\varphi(X,Y)$ is 0 on average. To summarize, we show that:

$$\mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{D}}[\varphi(X,Y)] = \mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{D}}[\varphi_{i\neq j}(X,Y)] \approx \mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{G}'}[\varphi_{i\neq j}(X,Y)] = O(\varepsilon^2)$$

Theorem 16. $\mathbf{E}_{(X,Y)\sim U_{2N}}[\varphi(X,Y)] = 0$.

Proof. For every $i, j, k \in [N]$, $\mathbf{E}_{(X,Y)\sim U_{2N}}[X_j X_k Y_i] = 0$ because Y_i is independent of X and has mean 0. By linearity of expectation $\mathbf{E}_{(X,Y)\sim U_{2N}}[\varphi(X,Y)] = 0$. □

Theorem 17. $\mathbf{E}_{(X,Y')\sim\mathcal{G}'}[\varphi_{i\neq j}(X,Y')] = \varepsilon^2\cdot(2 - 2/N)$.

Proof. By the definition of $\mathcal{G}'$ and $\varphi_{i\neq j}(X, Y')$ we can write:

$$\begin{aligned}
\mathop{\mathbf{E}}_{(X,Y')\sim\mathcal{G}'}[\varphi_{i\neq j}(X,Y')] &= \frac{1}{N}\sum_i\sum_{j\neq k} H_{ij}H_{ik}\,\mathbf{E}[X_jX_kY'_i] \\
&= \frac{1}{N}\sum_i\sum_{j\neq k} H_{ij}H_{ik}\,\mathbf{E}[X_jX_kY_i^2] - \varepsilon\,\mathbf{E}[X_jX_k] \\
&= \frac{1}{N}\sum_i\sum_{j\neq k} H_{ij}H_{ik}(\sigma'_{ii}\sigma'_{jk} + 2\sigma_{ij}\sigma_{ik}) \\
&= \frac{1}{N}\sum_i\sum_{j\neq k} H_{ij}^2H_{ik}^2 2\varepsilon^2 = \frac{2(N-1)}{N}\varepsilon^2
\end{aligned}$$

Where the third line follows from the fact that (X, Y) is a multivariate normal distribution, and the last inequality holds for $N \geq 2$. Furthermore, σ'_{ij} is the covariance of (X_i, X_j) and (Y_i, Y_j), and σ_{ij} is the covariance of (Y_i, X_j). □

Theorem 18. $\mathbf{E}_{(X,Y)\sim\mathcal{D}}[\varphi(X,Y)] \geq \varepsilon^2$.

Proof. Since $\mathbf{E}[Y_i] = 0$, by linearity of expectation we can write:

$$\begin{aligned}
\mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{D}}[\varphi(X,Y)] &= \frac{1}{N}\sum_i\sum_{j\neq k} H_{ij}H_{ik}\,\mathbf{E}[X_jX_kY_i] + \frac{1}{N}\sum_i\sum_j H_{ij}^2\,\mathbf{E}[X_j^2Y_i] \\
&= \frac{1}{N}\sum_i\sum_{j\neq k} H_{ij}H_{ik}\,\mathbf{E}[X_jX_kY_i] + \frac{1}{N}\sum_i\sum_j H_{ij}^2\,\mathbf{E}[Y_i] \\
&= \mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{D}}[\varphi_{i\neq j}(X,Y)]
\end{aligned}$$

Note that $\varphi_{i\neq j}(X,Y)$ is a multilinear function that takes $\{\pm 1\}^{2N}$ to $[-2, 2]$ since the subtracted quadratic part maps $\{\pm\}^{2N}$ to $[-1, 1]$ and $\varphi(X,Y) \in [-1, 1]$. So we can use Theorem 15 with $p_0 = 0, p = 1$ and Equation 1 to write:

$$\left| \mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{D}}[\varphi_{i\neq j}(X,Y)] - \mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{G}'}[\varphi_{i\neq j}(X,Y)] \right| \leq 16\cdot N^{-2}$$

Then we can get a lower bound on $\mathbf{E}_{(X,Y)\sim\mathcal{D}}[\varphi(X,Y)]$:

$$\mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{D}}[\varphi_{i\neq j}(X,Y)] \geq \mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{G}'}[\varphi_{i\neq j}(X,Y)] - 16\cdot N^{-2} \geq \varepsilon^2$$

□

3.3 Classical Hardness

Let $L_{1,2}(F) = \sum_{S\subseteq[2N],|S|=2} |\widehat{F}(S)|$. The results in this section combined with the tail bound from [Tal17], suffices to prove a lower bound on the size of any AC^0 circuit that distinguishes $\mathcal{D}$ from uniform.

Theorem 19 ([CHLT19, Claim A.5])**.** *Let f be a multi-linear function on $\mathbb{R}^N$ and $x \in [-1/2, 1/2]^N$. There exists a distribution over random restrictions $\mathcal{R}_x$ such that for any $y \in \mathbb{R}^N$:*

$$f(x+y) - f(x) = \mathop{\mathbf{E}}_{\rho\sim\mathcal{R}_x}[f_\rho(2\cdot y) - f_\rho(0)]$$

Theorem 20. *Let $t = N^{c-1}$ for a sufficiently large universal constant c (e.g., $c = 40$). Let $(X, Y) \in [-1/2, 1/2]^{2N}$, $\Delta X \sim \mathcal{N}(0, \varepsilon/t)$ and $\Delta Y = H \cdot \Delta X$. Let $F : \mathbb{R}^{2N} \to \mathbb{R}$ be a multilinear function where for all random restrictions $L_{1,2}(F_\rho) \leq \ell$:*

$$(*) = \left| \mathop{\mathbf{E}}_{\Delta X} \left[F\big(X + \Delta X, (Y + \Delta Y)^2 - \tfrac{i \cdot \varepsilon}{t}\big) \right] - \mathbf{E}\left[F\big(X, Y^2 - \tfrac{(i-1)\cdot\varepsilon}{t}\big) \right] \right| \leq \frac{O(\varepsilon \cdot \ell)}{t\sqrt{N}}$$

Proof. Let $Y' = Y^2 - \frac{\varepsilon\cdot(i-1)}{t}$. We can rewrite $(*)$ as follows:

$$(*) = \mathop{\mathbf{E}}_{\Delta X} \left[F(X + \Delta X, Y' + 2Y \cdot \Delta Y + \Delta Y^2 - \tfrac{\varepsilon}{t}) - F(X, Y') \right]$$

Note that $Y' \in [-1/2, 1/2]^N$. Using Theorem 19 we can write this value using random restrictions of F:

$$(*) = \mathop{\mathbf{E}}_{\Delta X, \rho} \left[F_\rho\big(2 \cdot \Delta X, 2 \cdot \Delta Y'\big) - F_\rho(0, 0) \right]$$

Where $\Delta Y' = 2Y \cdot \Delta Y + \Delta Y^2 - \frac{\varepsilon}{t}$. Let $Z = (2\Delta X, 2\Delta Y')$. Next we can expand $(*)$ in terms of multilinear expansion of F_ρ. From the Isserlis' theorem [Iss18] we can see that monomials of degree $k > 2$ in the variables Z scale like $O(\frac{k^k}{t^{k/2}})$, and there are at most N^k such monomials. Since we can choose t to be arbitrarily large power of N, their contribution would be negligible, i.e., $o(\varepsilon/t\sqrt{N})$. Furthermore, since for every input variable X_i of F_ρ, $\mathbf{E}[Z_i] = 0$ (using the fact that $\mathbf{E}[\Delta Y^2 - \frac{\varepsilon}{t}] = 0$) we can also ignore monomials of degree 1. Note that this is precisely why we also break ε in the telescopic sum. The remaining step is to bound monomials of degree 2.

To account for monomials of degree 2, we upper bound the covariances of all pairs of variables from $Z = (2\Delta X, 2\Delta Y')$ by $O(\frac{\varepsilon}{t\sqrt{N}})$. First, for $i \neq j$ we have that $(\Delta X)_i$ and $(\Delta X)_j$ are independent and thus have covariance 0. Second, we bound $\left|\mathbf{Cov}\left(2\left(\Delta X\right)_i, 2\left(\Delta Y'\right)_j\right)\right|$ for any (i, j) by:

$$|4 \cdot 2Y_j \, \mathbf{Cov}((\Delta X)_i, (\Delta Y)_j) + 4 \cdot \mathbf{Cov}((\Delta X)_i, (\Delta Y)_j^2)| \leq \frac{4\varepsilon}{t\sqrt{N}} + O\left(\tfrac{1}{t^{3/2}}\right) = O\left(\frac{\varepsilon}{t\sqrt{N}}\right)$$

Similarly, we can bound $|\mathbf{Cov}\,(2(\Delta Y')_i, 2(\Delta Y')_j)|$ for $i \neq j$

$$4 \cdot 4Y_iY_j \, \mathbf{Cov}\left((\Delta Y)_i, (\Delta Y)_j\right) + O\left(\frac{1}{t^{3/2}}\right) \leq O\left(\frac{\varepsilon}{t\sqrt{N}}\right)$$

Finally, we can bound $(*)$:

$$\begin{aligned}\left| \mathop{\mathbf{E}}_{\Delta X, \rho} [F_\rho(Z) - F_\rho(0^{2N})] \right| &\leq \sum_{i,j} \left| \mathop{\mathbf{E}}_{\Delta X, \rho} [\widehat{F_\rho}(\{i, j\}) \cdot Z_i Z_j] \right| + o\left(\tfrac{\varepsilon\ell}{t\sqrt{N}}\right) \\ &\leq \max_{i,j} |\mathbf{Cov}(Z_i, Z_j)| \cdot \max_\rho \sum_{\substack{S \subseteq [2N], \\ |S| = 2}} |\widehat{F_\rho}(S)| + o\left(\tfrac{\varepsilon\ell}{t\sqrt{N}}\right) \\ &\leq O\left(\frac{\varepsilon\ell}{t\sqrt{N}}\right).\end{aligned}$$

□

Theorem 21. *Let $\ell \geq 1$. Let $F : \{\pm 1\}^{2N} \to \{\pm 1\}$ be a multilinear function with bounded spectral norm $L_{1,2}(F) \leq \ell$. Then:*

$$(*) = \mathop{\mathbf{E}}_{(X, Y') \sim \mathcal{D}} \left[F(X, Y') - F(0, 0) \right] \leq O\left(\frac{\varepsilon \cdot \ell}{\sqrt{N}}\right)$$

Proof. From Equation 1, it suffices to show:

$$\mathop{\mathbf{E}}_{(X,Y)\sim\mathcal{G}}\left[F\big(\operatorname{trnc}(X),\operatorname{trnc}(Y^2-\varepsilon)\big)-F(0,0)\right]\le O\left(\frac{\varepsilon\cdot\ell}{\sqrt{N}}\right)$$

We start by writing this value as a telescoping sum. Let $t = N^{c-1}$ for c the constant guaranteed by Theorem 20. Set C in the definition of ε to guarantee that the error in Theorem 15 is at most $8\cdot N^{-c}$ and that $e^{-1/(8\varepsilon)} \le N^{-(c+1)}$. Then we can write:

$$(*) = \sum_{i=1}^{t}\mathbf{E}\left[F(\operatorname{trnc}(X^{(i)}),\operatorname{trnc}(Y^{(i)^2}-\tfrac{\varepsilon\cdot i}{t}))-F(\operatorname{trnc}(X^{(i-1)}),\operatorname{trnc}(Y^{(i-1)^2}-\tfrac{\varepsilon\cdot(i-1)}{t}))\right]$$

where $X^{(i)} = X^{(i-1)} + \Delta X$, $Y^{(i)} = Y^{(i-1)} + \Delta Y$ and $\Delta X \sim \mathcal{N}(0,\varepsilon/t)$ and $\Delta Y = H\cdot\Delta X$. We are going to use Theorem 8 to bound each term in the sum to get the final result. Let E_{i-1} be the event that $(X^{(i-1)}, Y^{(i-1)} - \frac{\varepsilon\cdot(i-1)}{t}) \in [-1/2, 1/2]^{2N}$. We can prove that E_{i-1} happens with high probability:

$$\mathbf{Pr}[(X^{(i-1)},Y^{(i-1)})\notin[-1/2,1/2]^{(2N)}]\le 2N\cdot\mathbf{Pr}[|\mathcal{N}(0,\varepsilon)|\ge 1/2]\le 2N\cdot 2e^{-1/(8\varepsilon)}\le 4N^{-c}$$

It is also clear that if $(X^{(i-1)}, Y^{(i-1)}) \in [-1/2, 1/2]^{2N}$ then $(X^{(i-1)}, Y^{(i-1)^2} - \frac{\varepsilon\cdot(i-1)}{t}) \in [-1/2, 1/2]^{2N}$. So $\mathbf{Pr}[E_{i-1}] \ge 1 - 4N^{-c}$. Next we just have to use Theorems 15 and 20 combined with triangle inequality to prove the desired claim. We can bound the i-th term in the sum, S_i, by:

$$\begin{aligned}S_i &\le \mathbf{E}\left[F(X^{(i)},Y^{(i)^2}-\tfrac{\varepsilon\cdot i}{t})-F(X^{(i-1)},Y^{(i-1)^2}-\tfrac{\varepsilon\cdot(i-1)}{t})\middle|E_{i-1}\right]\\&+\mathbf{E}\left[F(\operatorname{trnc}(X^{(i)}),\operatorname{trnc}(Y^{(i)^2}-\tfrac{\varepsilon\cdot i}{t}))-F(X^{(i)},Y^{(i)^2}-\tfrac{\varepsilon\cdot i}{t})\middle|E_{i-1}\right]\\&+2\,\mathbf{Pr}[\neg E_{i-1}]\le O\left(\frac{\varepsilon\cdot\ell}{t\sqrt{N}}\right)+O(N^{-c})\end{aligned}$$

So we can bound the sum $(*) \le O(\frac{\varepsilon\cdot\ell}{\sqrt{N}}) + O(tN^{-c}) \le O(\frac{\varepsilon\cdot\ell}{\sqrt{N}})$. □

Remark 22. *Note that this proof can easily be extended to the "long list" version of problem, where we are given oracle access to a list of pairs of functions (f_i, g_i) rather than a single pair which has been shown. We have already shown this for Theorem 14 and 15 in Appendix A, which directly prove Theorem 20. The only remaining step that we need to verify in Theorem 21 is the probability of the event E_i. Similar to what has been shown in Appendix A, we can choose ε to be small enough so that $p(N)\cdot e^{-1/(8\varepsilon)}$ remains exponentially small, and the rest of the proof follows immediately. Later we will use this result to give evidence for* LLQSV*.*

3.4 More details on the relation between SquaredForrelation and Long List

As we have mentioned before, our goal is to use samples $(f, g) \sim \mathcal{D}$, and then use g to get samples according to the Fourier transform distribution. However, if we choose a uniformly random sample such that $g(x) = 1$, we would not get a sample *exactly* according to the Fourier transform distribution. Intuitively, this is a "flattened version" of the Fourier transform distribution of f, but we can still prove a PH lower bound for this variant of black-box LLQSV, where the samples are taken according to this flattened distribution. We show that this distinction does not matter, by first noting that we can define a modified HOG score – Rejection Sampling HOG (RHOG) – so that RHOG and the flattened distribution have the same relation as HOG and black-box LLQSV. Furthermore, we prove that a quantum algorithm can still solve both HOG and RHOG by sampling according to the Fourier transform distribution of f. Let $\mathcal{D}$ be the distribution defined in the previous section, and let $\mathcal{D}_f$ be the marginal distribution over g for a fixed f. Given a random Boolean function f, $\mathcal{R}_f$ samples $s \in \{0,1\}^n$ as follows:

- Sample $g \sim \mathcal{D}_f$.
- Sample $x \in \{0,1\}^n$ uniformly at random and output x if $g(x) = 1$.
- Output a random element after $4n^2$ unsuccessful attempts.

Suppose we are given access to oracle $\mathcal{O}$ which is a long (exponentially large) list of random Boolean functions $f_1, f_2, \ldots, f_T : \{\pm 1\}^n \to \{0,1\}$ paired with strings $s_1, s_2, \ldots, s_T$. Here we define black-box LLQSV to be the problem of distinguishing samples from $\mathcal{R}_f$ from uniform. One can hope that since g is correlated with squared Fourier coefficients of f, these samples are also close to the "Fourier distribution". In fact, we later show that on average (over the random Boolean function and the samples), $\mathcal{R}_f$ generates heavy Fourier coefficients. The same argument also works to prove that an honest quantum sampler, that samples according the Fourier coefficients, passes the following score:

Problem 23. (RHOG) *Let $b = 1 + \mathrm{poly}(1/n)$. Given a random Boolean function $f : \{0,1\}^N \to \{\pm 1\}$, output an outcome $s \in \{0,1\}^N$ such that:*

$$\mathop{\mathbf{E}}_{f,s}\left[\mathbf{Pr}\left[\mathcal{R}_f \textit{ samples } s\right]\right] \geq \frac{b}{N}$$

One can easily verify that the existence of a deterministic quantum algorithm for solving RHOG also gives a QCAM algorithm for (modified) black-box LLQSV. This follows from a collision finding algorithm similar to the original LLQSV argument. Here we prove that first, the usual quantum algorithm that samples according to the Fourier distribution passes solves RHOG. Next, we show that SquaredForrelation can be reduced to black-box LLQSV, which implies that black-box LLQSV is not in PH.

We prove the first result by using two properties of the distribution $\mathcal{G}'$. The first directly follows from a tail bound of Chi-square variables:

Lemma 24 ([LM00]). *Let $(Y_1, \ldots, Y_N) \sim \mathcal{N}(0,1)$ and let $a_1, \ldots, a_N \geq 0$. Let*

$$Z = \sum_i a_i (Y_i^2 - 1)$$

Then for any $\Delta > 0$ we have:

- $\mathbf{Pr}\left[Z \geq 2|a|_2\sqrt{\Delta} + 2|a|_\infty \Delta\right] \leq \exp(-\Delta)$
- $\mathbf{Pr}\left[Z \leq -2|a|_2\sqrt{\Delta}\right] \leq \exp(-\Delta)$

Theorem 25. *Let $(X, Y) \sim \mathcal{G}'$, then* $\mathbf{Pr}\left[\left|\sum_i Y_i\right| \geq 3\sqrt{N}\right] \leq 2\exp(-1/\varepsilon)$.

Proof. Using Lemma 24 by letting $a = (\varepsilon, \ldots, \varepsilon)$ and $\Delta = \frac{1}{\varepsilon}$ we get:

- $\mathbf{Pr}\left[Z \geq 2\sqrt{N} + 2\right] \leq \exp(-\frac{1}{\varepsilon})$
- $\mathbf{Pr}\left[Z \leq -2\sqrt{N}\right] \leq \exp(-\frac{1}{\varepsilon})$

And a union bound gives us the claim. □

Recall that for $(X,Y) \in \mathcal{G}'$ with high probability we have $\mathrm{trnc}(X,Y) = (X,Y)$:

Theorem 26. *Let $(X,Y) \sim \mathcal{G}'$, then $\mathbf{Pr}\left[\mathrm{trnc}(X,Y) \neq (X,Y)\right] \leq 2N^{-2}$.*

Corollary 27. *Let $(X,Y) \sim \mathcal{D}$ and $\delta = N^{-1/3}$, then with probability at least $1-5N^{-2}$ we have:*

$$(1-\delta)\frac{N}{2} \leq \left|\{i|Y_i = 1\}\right| \leq (1+\delta)\frac{N}{2}$$

Proof. This follows from a Chernoff bound applied to Theorem 25 and 26. Given $(X',Y') \sim \mathcal{G}'$, by a union bound we get that the probability that both requirements $\mathrm{trnc}(X',Y') = (X',Y')$ and $|\sum Y_i'| < 3\sqrt{N}$ are satisfied is at least $1-4N^{-2}$. In such a case, recall that Y is attained from Y' by randomized rounding (without truncations since $Y' = \mathrm{trnc}(Y')$. Let $\delta' = \delta/2$. Then by Chernoff bound, with probability $1-\exp(\Omega(\delta^2 N)) \geq 1-N^{-2}$:

$$|\{i|Y_i = 1\}| \leq (1+\delta')\left(\frac{N}{2} + 3\sqrt{N}\right) \leq (1+\delta)\frac{N}{2}$$

and,

$$|\{i|Y_i = 1\}| \geq (1-\delta')\left(\frac{N}{2} - 3\sqrt{N}\right) \geq (1-\delta)\frac{N}{2}$$

for large enough N. This means that the claim holds over $\mathcal{D}$ with probability at least $1-5N^{-2}$. □

Another direct consequence of this property is that we can bound the success probability of getting a non-uniform sample after $4n^2$ steps in rejection sampling. Note that the probability of the rejection sampling failing is bounded by $\left((1+\delta)/2\right)^{4n^2}$. For large enough N and $\delta = 1/N^{1/3}$, we know $((1+\delta)/2)^2 \leq 1/2$, and thus

$$((1+\delta)/2)^{4n^2} \leq 2^{-2n^2} \ll N^{-2}$$

Now we can prove that given random Boolean function $f : \{\pm 1\}^N \to \{0,1\}$, sampling according to the Fourier coefficients of f solves RHOG. Define $\mathcal{C}_f$ to be the Fourier transform distribution over outcomes according to f, i.e., $\mathcal{C}_f$ samples x with probability $\widehat{f}(x)^2$.

Claim 28.

$$\mathop{\mathbf{E}}_{\substack{f,\\ x\sim\mathcal{C}_f}}\left[\mathbf{Pr}\left[\mathcal{R}_f \text{ samples } x\right]\right] \geq \frac{1+\mathrm{poly}(\varepsilon)}{N}$$

Proof. From Theorem 18, we know that $\mathbf{E}_{\mathcal{D}}\left[\sum_{g(x)=1} \widehat{f}(x)^2\right] \geq \frac{1}{2} + \frac{\varepsilon^2}{4}$. Let E be the event that :

1. $(1-\delta)\frac{N}{2} \leq \left|\{i|Y_i = 1\}\right| \leq (1+\delta)\frac{N}{2}$
2. The rejection sampling algorithm outputs an s such that $g(s) = 1$.

where $\delta = N^{-1/3}$. We know that $\mathbf{Pr}[E] \geq 1 - 10/N^2$ and $\mathbf{E}_{\mathcal{D}}\left[\sum_{g(x)=1} \widehat{f}(x)^2 | E\right] \geq \frac{1}{2} + \frac{\varepsilon^2}{8}$. We can lower bound

$$\mathop{\mathbf{E}}_{\substack{f,\\ x\sim\mathcal{C}_f}}\left[\mathbf{Pr}\left[\mathcal{R}_f \text{ samples } x\right]\right] \geq \mathbf{Pr}[E] \cdot \mathop{\mathbf{Pr}}_{\substack{f,\\ x\sim\mathcal{C}_f}}\left[\mathcal{R}_f \text{ samples } x | E\right].$$

Note further that conditioned on E, the rejection sampling algorithm draws a random element from the set $\{y : g(y) = 1\}$. Thus we may rewrite

$$\begin{aligned}\mathbf{Pr}[E] \cdot \underset{\substack{f,\\ x\sim\mathcal{C}_f}}{\mathbf{Pr}}[\mathcal{R}_f \text{ samples } x|E] &= \mathbf{Pr}[E] \cdot \underset{(f,g)\sim\mathcal{D}}{\mathbf{E}}\left[\frac{\sum_{g(x)=1}\widehat{f}(x)^2}{|\{y : g(y) = 1\}|} \,\middle|\, E\right] \\ &\geq \mathbf{Pr}[E] \cdot \frac{1/2 + \varepsilon^2/8}{(1+\delta)N/2} \\ &\geq \mathbf{Pr}[E] \cdot \frac{(1-\delta)(1+\varepsilon^2/4)}{N}\end{aligned}$$

which is at least $\frac{1+\varepsilon^2/8}{N}$ for N large enough (where we used the fact that $\varepsilon = \Theta(1/\log N)$ is much larger than $10/N^2$ and δ). □

The only remaining step is to prove hardness of (modified) black-box LLQSV. First we define the following problem:

Problem 29. *Given oracle access to a length $T = 2^{3n}$ list of triplets (f_i, g_i, s_i) where $f_i, g_i : \{\pm 1\}^N \to \{0,1\}$ and $s_i \in \{0,1\}^N$ is sampled according to the rejection sampling algorithm applied to g_i distinguish between the following two cases:*

1. *For every i, f_i, g_i are chosen uniformly at random.*
2. *For every i. f_i, g_i are sampled according to $\mathcal{D}$.*

We can use this to prove our final result:

Theorem 30. *No constant depth Boolean circuit of size* $\text{quasipoly}(N)$ *can solve black-box* LLQSV *with advantage more than* $\text{polylog}(N)/\sqrt{N}$.

Proof. First, it is clear that we can reduce Problem 29 to black-box LLQSV since we have access to the list (f_i, s_i). To prove the same oracle separation result using Problem 29, we first notice that we can prove the same $\mathsf{AC^0}$ lower bound if we only had access to (f_i, g_i), but not s_i (Remark 22). Furthermore, by choosing ε in the definition of $\mathcal{D}$ small enough, the lower bound even holds for a list of length $2 \cdot 2^{n^3}$. The final step is to prove that s_i's do not give us extra computation power. Suppose there exist a circuit C of size s and depth d for Problem 29, we can convert this circuit to work without having access to s_i. First, we can replace s_i with a small circuit that given $r_i \in \{0,1\}^{\text{poly}(n)}$, the random seed of the rejection sampling, simulate rejection sampling, making $O(n^2)$ queries to g_i and outputs s_i. Such a computation can be described by a decision tree on that reads r_i and at most $O(n^2)$ entries of g_i, and thus also by a DNF of size $2^{\text{poly}(n)} = \text{quasipoly}(N)$. Making all these replacements everywhere gives a new circuit with size $s' = s \cdot \text{quasipoly}(N)$ (which is still quasi-polynomial in N) and depth $d+2$. Furthermore, if we have access to a list of length $2 \cdot 2^{n^3}$ of pairs of functions, we can use the second half as the random seed, instead of accessing r_i directly. So this new circuit of size s' and depth $d+2$ must also distinguish the two distribution with the same advantage, having only access to a list of length $2 \cdot 2^{n^3}$ of (f_i, g_i). Thus, the same lower bound applies to the size of the starting circuit s. □

Combining Theorem 30 with standard conversion between PH and $\mathsf{AC^0}$ proves a second PH bound of Theorem 3 for black-box LLQSV.

References

[AA13a] Scott Aaronson and Alex Arkhipov. Bosonsampling is far from uniform. *arXiv preprint arXiv:1309.7460*, 2013.

[AA13b] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. *Theory Comput.*, 9:143–252, 2013.

[AA18] Scott Aaronson and Andris Ambainis. Forrelation: A problem that optimally separates quantum from classical computing. *SIAM J. Comput.*, 47(3):982–1038, 2018.

[AAB$^+$19] Frank Arute, Kunal Arya, Ryan Babbush, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[Aar10] Scott Aaronson. BQP and the polynomial hierarchy. In *STOC*, pages 141–150, 2010.

[Aar18a] Scott Aaronson. Certified randomness from quantum supremacy. *Talk at CRYPTO 2018*, October 2018.

[Aar18b] Scott Aaronson. Quantum supremacy and its applications. *Talk at the Simons Institute for the Theory of Computing*, June 2018.

[Aar20] Scott Aaronson. On entropy from random circuit sampling. *Talk at the Simons Institute for the Theory of Computing*, May 2020.

[AC17] Scott Aaronson and Lijie Chen. Complexity-theoretic foundations of quantum supremacy experiments. In Ryan O'Donnell, editor, *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, volume 79 of *LIPIcs*, pages 22:1–22:67. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[AG19] Scott Aaronson and Sam Gunn. On the classical hardness of spoofing linear cross-entropy benchmarking. *CoRR*, abs/1910.12085, 2019.

[AM16] Antonio Acín and Lluis Masanes. Certified randomness in quantum physics. *Nature*, 540(7632):213–219, 2016.

[BBBV97] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh V. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997.

[BCM$^+$18] Zvika Brakerski, Paul Christiano, Urmila Mahadev, et al. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 320–331. IEEE, 2018.

[BFLL21] Adam Bouland, Bill Fefferman, Zeph Landau, and Yunchao Liu. Noise and the frontier of quantum supremacy. *CoRR*, abs/2102.01738, 2021.

[BFNV19] Adam Bouland, Bill Fefferman, Chinmay Nirkhe, and Umesh Vazirani. On the complexity and verification of quantum random circuit sampling. *Nature Physics*, 15(2):159–163, 2019.

[BHZ87] Ravi B. Boppana, Johan Håstad, and Stathis Zachos. Does co-NP have short interactive proofs? *Inf. Process. Lett.*, 25(2):127–132, 1987.

[BIS^{+}18] Sergio Boixo, Sergei V Isakov, Vadim N Smelyanskiy, et al. Characterizing quantum supremacy in near-term devices. *Nature Physics*, page 1, 2018.

[BJS11] 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: Mathematical, Physical and Engineering Sciences*, 467(2126):459–472, 2011.

[BKVV20] Zvika Brakerski, Venkata Koppula, Umesh V. Vazirani, and Thomas Vidick. Simpler proofs of quantumness. *CoRR*, abs/2005.04826, 2020.

[BMS16] Michael J Bremner, Ashley Montanaro, and Dan J Shepherd. Average-case complexity versus approximate simulation of commuting quantum computations. *Physical review letters*, 117(8):080501, 2016.

[BV97] Ethan Bernstein and Umesh V. Vazirani. Quantum complexity theory. *SIAM J. Comput.*, 26(5):1411–1473, 1997.

[CHHL19] Eshan Chattopadhyay, Pooya Hatami, Kaave Hosseini, and Shachar Lovett. Pseudorandom generators from polarizing random walks. *Theory Comput.*, 15:1–26, 2019.

[CHLT19] Eshan Chattopadhyay, Pooya Hatami, Shachar Lovett, and Avishay Tal. Pseudorandom generators from the second fourier level and applications to AC0 with parity gates. In Avrim Blum, editor, *10th Innovations in Theoretical Computer Science Conference, ITCS 2019, January 10-12, 2019, San Diego, California, USA*, volume 124 of *LIPIcs*, pages 22:1–22:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[CSV21] Matthew Coudron, Jalex Stark, and Thomas Vidick. Trading locality for time: Certifiable randomness from low-depth circuits. *Communications in Mathematical Physics*, 382(1):49–86, February 2021.

[FSUV13] Bill Fefferman, Ronen Shaltiel, Christopher Umans, and Emanuele Viola. On beating the hybrid argument. *Theory Comput.*, 9:809–843, 2013.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 212–219, 1996.

[GUV09] Venkatesan Guruswami, Christopher Umans, and Salil P. Vadhan. Unbalanced expanders and randomness extractors from parvaresh-vardy codes. *J. ACM*, 56(4):20:1–20:34, 2009.

[Hås87] Johan Håstad. *Computational Limitations of Small-Depth Circuits.* MIT Press, Cambridge, MA, USA, 1987.

[HG21] Shuichi Hirahara and François Le Gall. Test of quantumness with small-depth quantum circuits. In Filippo Bonchi and Simon J. Puglisi, editors, *46th International Symposium on Mathematical Foundations of Computer Science, MFCS 2021, August 23-27, 2021, Tallinn, Estonia*, volume 202 of *LIPIcs*, pages 59:1–59:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[HKEG19] Dominik Hangleiter, Martin Kliesch, Jens Eisert, and Christian Gogolin. Sample complexity of device-independently certified "quantum supremacy". *Physical Review Letters*, 122(21), May 2019.

[Iss18] Leon Isserlis. On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables. *Biometrika*, 12(1/2):134–139, 1918.

[KCVY21] Gregory D. Kahanamoku-Meyer, Soonwon Choi, Umesh V. Vazirani, and Norman Y. Yao. Classically-verifiable quantum advantage from a computational bell test. *CoRR*, abs/2104.00687, 2021.

[KMM21] Yasuhiro Kondo, Ryuhei Mori, and Ramis Movassagh. Improved robustness of quantum supremacy for random circuit sampling, 2021.

[LG21] Zhenning Liu and Alexandru Gheorghiu. Depth-efficient proofs of quantumness. *arXiv:2107.02163 [quant-ph]*, July 2021.

[LM00] B. Laurent and P. Massart. Adaptive estimation of a quadratic functional by model selection. *The Annals of Statistics*, 28(5):1302–1338, 2000.

[Mov19] Ramis Movassagh. Cayley path and quantum computational supremacy: A proof of average-case #p-hardness of random circuit sampling with quantified robustness. *CoRR*, abs/1909.06210, 2019.

[PAM$^+$10] S. Pironio, A. Acín, S. Massar, et al. Random numbers certified by bell's theorem. *Nature*, 464(7291):1021–1024, Apr 2010.

[PT56] C. E. Porter and R. G. Thomas. Fluctuations of nuclear reaction widths. *Phys. Rev.*, 104:483–491, Oct 1956.

[RT19] Ran Raz and Avishay Tal. Oracle separation of BQP and PH. In Moses Charikar and Edith Cohen, editors, *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*, pages 13–23. ACM, 2019.

[Sho94] Peter W. Shor. Polynominal time algorithms for discrete logarithms and factoring on a quantum computer. In *ANTS*, page 289, 1994.

[Tal17] Avishay Tal. Tight bounds on the Fourier spectrum of AC0. In Ryan O'Donnell, editor, *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, volume 79 of *LIPIcs*, pages 15:1–15:31. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[TD04] Barbara M Terhal and David P DiVincenzo. Adptive quantum computation, constant depth quantum circuits and Arthur-Merlin games. *Quantum Information and Computation*, 4(2):134–145, 2004.

[Vio07] Emanuele Viola. On approximate majority and probabilistic time. In *22nd Annual IEEE Conference on Computational Complexity (CCC 2007), 13-16 June 2007, San Diego, California, USA*, pages 155–168. IEEE Computer Society, 2007.

[VV17] Gregory Valiant and Paul Valiant. An automatic inequality prover and instance optimal identity testing. *SIAM J. Comput.*, 46(1):429–455, 2017.

[Wu18] Xinyu Wu. A stochastic calculus approach to the oracle separation of BQP and PH. *Electron. Colloquium Comput. Complex.*, page 202, 2018.

[ZCC+21] Qingling Zhu, Sirui Cao, Fusheng Chen, et al. Quantum computational advantage via 60-qubit 24-cycle random circuit sampling, 2021.

[ZDQ+21] Han-Sen Zhong, Yu-Hao Deng, Jian Qin, et al. Phase-programmable gaussian boson sampling using stimulated squeezed light, 2021.

[ZWD+20] Han-Sen Zhong, Hui Wang, Yu-Hao Deng, et al. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, Dec 2020.

A Truncated Gaussians - Proofs

Here we prove a long list version of Theorems 14 and 15. The core statements are the same, but the expectation is over the distribution $\mathcal{G}'^{p(N)}$ for some polynomial p. In LLQSV we are interested in the case where $p(N) = N^3$. We show that for any polynomial p we can always choose $\varepsilon = 1/(C \ln N)$, to make expectation values in the next theorems arbitrarily small (N^{-c} for some constant c that depends on p and C).

Proof of Thm. 14. We can write:

$$\begin{aligned}
(*) &= \mathop{\mathbf{E}}_{(x,y)\sim\mathcal{G}'^{p(n)}}\left[\prod_{i=1}^{p(N)\cdot N}\max\left(1,|x_i|\right)\cdot\prod_{i=1}^{p(N)\cdot N}\max\left(1,|y_i|\right)\cdot\mathbb{1}_{(x,y)\neq\mathrm{trnc}(x,y)}\right]\\
&= \mathop{\mathbf{E}}_{(x,y)\sim\mathcal{G}'^{p(n)}}\left[\prod_{i=1}^{p(N)\cdot N}\max\left(1,|x_i|\right)\cdot\prod_{i=1}^{p(N)\cdot N}\max\left(1,|y_i|\right)\right]\\
&\quad - \mathop{\mathbf{E}}_{(x,y)\sim\mathcal{G}'^{p(n)}}\left[\prod_{i=1}^{p(N)\cdot N}\max\left(1,|x_i|\right)\cdot\prod_{i=1}^{p(N)\cdot N}\max\left(1,|y_i|\right)\cdot\mathbb{1}_{(x,y)=\mathrm{trnc}(x,y)}\right]\\
&= \mathop{\mathbf{E}}_{(x,y)\sim\mathcal{G}'^{p(n)}}\left[\prod_{i=1}^{p(N)\cdot N}\max\left(1,|x_i|\right)\cdot\prod_{i=1}^{p(N)\cdot N}\max\left(1,|y_i|\right)\right] - \mathbf{Pr}\left[(x,y)=\mathrm{trnc}(x,y)\right]
\end{aligned}$$

Let $z \coloneqq (x, y)$ and $\varepsilon = \frac{1}{2k\ln(p(N)\cdot N)}$ for some constant k. We first bound the second term:

$$\mathbf{Pr}\left[z=\mathrm{trnc}(z)\right] \geq 1-\sum_i \mathbf{Pr}[z\neq\mathrm{trnc}(z)] = 1-2N\cdot p(N)\cdot e^{-1/(2\varepsilon)} \geq 1-N^{-k+1}$$

Next we can bound the first term using Cauchy-Schwarz:

$$\begin{aligned}
&\leq \sqrt{\mathop{\mathbf{E}}_{x}\left[\prod_{i=1}^{p(N)\cdot N}\max\left(1,x_i^2\right)\right]}\cdot\sqrt{\mathbb{E}_x\left[\prod_{i=1}^{p(N)\cdot N}\max\left(1,(x_i^2-\varepsilon)^2\right)\right]}\\
&= \mathop{\mathbf{E}}_{x}\left[\max\left(1,x_i^2\right)\right]^{p(N)\cdot N/2}\cdot\mathop{\mathbf{E}}_{x}\left[\max\left(1,(x_i^2-\varepsilon)^2\right)\right]^{p(N)\cdot N/2}
\end{aligned}$$

We can write the first expectation value as:

$$\begin{aligned}2\int_0^1 \frac{1}{\sqrt{2\pi\varepsilon}}e^{-x^2/2\varepsilon}dx + 2\int_1^\infty \frac{x^2}{\sqrt{2\pi\varepsilon}}e^{-x^2/2\varepsilon}dx &\le 1+\varepsilon\cdot\mathrm{erfc}(1/\sqrt{2\varepsilon}) + \sqrt{2\varepsilon/\pi}\cdot e^{-1/2\varepsilon}\\ &\le 1+\varepsilon e^{-1/2\varepsilon} + \sqrt{2\varepsilon/\pi}e^{-1/2\varepsilon}\\ &\le 1+N^{-k+1}\cdot p(N)^{-k}\end{aligned}$$

For the second expectation value:

$$\begin{aligned}&2\int_0^{\sqrt{1+\varepsilon}} \frac{1}{\sqrt{2\pi\varepsilon}}e^{-x^2/2\varepsilon}dx + 2\int_{\sqrt{1+\varepsilon}}^\infty \frac{(x^2-\varepsilon)^2}{\sqrt{2\pi\varepsilon}}e^{-x^2/2\varepsilon}dx\\ &\le 2\int_0^{\sqrt{1+\varepsilon}} \frac{1}{\sqrt{2\pi\varepsilon}}e^{-x^2/2\varepsilon}dx + 2\int_1^\infty \frac{x^4}{\sqrt{2\pi\varepsilon}}e^{-x^2/2\varepsilon}\\ &\le 1+3\varepsilon^2\,\mathrm{erfc}(1/\sqrt{2\varepsilon}) + \frac{\varepsilon(1+3\varepsilon)}{\sqrt{2\pi\varepsilon}}e^{-1/2\varepsilon}\\ &\le 1+3\varepsilon^2 e^{1/2\varepsilon} + \frac{\varepsilon(1+3\varepsilon)}{\sqrt{2\pi\varepsilon}}e^{-1/2\varepsilon} \le 1+N^{-k+1}\cdot p(N)^{-k}\end{aligned}$$

So we can bound the first term by $(1+N^{-k+1}\cdot p(N)^{-k})^{N\cdot p(N)} \le 1+N^{-k+2}$. Combining both we get the claim:

$$(*) \le 1+N^{-k+2} - (1-N^{-k+1}) \le 2N^{-k+2}$$

□

Proof of Thm. 15. The original proof works without any change with the modified Claim 5.2.

$$\begin{aligned}&\mathop{\mathbf{E}}_{z\sim\mathcal{G}'^{p(N)}}\left[\left|F\left(\mathrm{trnc}\left(z_0+p\cdot z\right)\right) - F\left(z_0+p\cdot z\right)\right|\right]\\ &\le \mathop{\mathbf{E}}_{z\sim\mathcal{G}'^{p(N)}}\left[\left(1+\left|F\left(z_0+p\cdot z\right)\right|\right)\cdot \mathbb{1}_{\mathrm{trnc}(z_0+p\cdot z)\neq z_0+p\cdot z}\right]\\ &\le \mathop{\mathbf{E}}_{z\sim\mathcal{G}'^{p(N)}}\left[\left(1+\left|F\left(z_0+p\cdot z\right)\right|\right)\cdot \mathbb{1}_{z\neq\mathrm{trnc}(z)}\right]\\ &\le \mathop{\mathbf{E}}_{z\sim\mathcal{G}'^{p(N)}}\left[\left(1+\prod_{i=1}^{2N\cdot p(N)}\max\left(1,\left|\left(z_0\right)_i+p\cdot z_i\right|\right)\right)\cdot \mathbb{1}_{z\neq\mathrm{trnc}(z)}\right]\\ &\le \mathop{\mathbf{E}}_{z\sim\mathcal{G}'^{p(N)}}\left[2\cdot\prod_{i=1}^{2N\cdot p(N)}\max\left(1,\left|\left(z_0\right)_i+p\cdot z_i\right|\right)\cdot \mathbb{1}_{z\neq\mathrm{trnc}(z)}\right]\end{aligned}$$

And since we know $\prod_{i=1}^{2N\cdot p(N)}\max\left(1,\left|\left(z_0\right)_i+p\cdot z_i\right|\right) \le \prod_{i=1}^{2N\cdot p(N)}\max\left(1,p_0+p\left|z_i\right|\right) \le \prod_{i=1}^{2N\cdot p(N)}\max\left(1,\left|z_i\right|\right)$, we can write:

$$\begin{aligned}&\mathop{\mathbf{E}}_{z\sim\mathcal{G}'^{p(N)}}\left[\left|F\left(\mathrm{trnc}\left(z_0+p\cdot z\right)\right) - F\left(z_0+p\cdot z\right)\right|\right]\\ &\quad\le \mathop{\mathbf{E}}_{z\sim\mathcal{G}'^{p(N)}}\left[2\cdot\prod_{i=1}^{2N\cdot p(N)}\max\left(1,\left|z_i\right|\right)\cdot \mathbb{1}_{z\neq\mathrm{trnc}(z)}\right] \le 4\cdot N^{-k+2}.\end{aligned}$$

□