On the Complexity and Verification of Quantum Random Circuit Sampling

Adam Bouland
Electrical Engineering and Computer Sciences, University of California, Berkeley
`abouland@berkeley.edu`

Bill Fefferman[*]
Electrical Engineering and Computer Sciences, University of California, Berkeley
Joint Center for Quantum Information and Computer Science (QuICS), University of Maryland / NIST
`wjf@berkeley.edu`

Chinmay Nirkhe
Electrical Engineering and Computer Sciences, University of California, Berkeley
`nirkhe@cs.berkeley.edu`

Umesh Vazirani
Electrical Engineering and Computer Sciences, University of California, Berkeley
`vazirani@cs.berkeley.edu`

Abstract

A critical milestone on the path to useful quantum computers is the demonstration of a quantum computation that is prohibitively hard for classical computers – a task referred to as quantum supremacy. A leading near-term candidate is sampling from the probability distributions of randomly chosen quantum circuits, which we call Random Circuit Sampling (RCS).

RCS was defined with experimental realizations in mind, leaving its computational hardness unproven. Here we give strong complexity-theoretic evidence of classical hardness of RCS, placing it on par with the best theoretical proposals for supremacy. Specifically, we show that RCS satisfies an average-case hardness condition, which is critical to establishing computational hardness in the presence of experimental noise. In addition, it follows from known results that RCS also satisfies an anti-concentration property, namely that errors in estimating output probabilities are small with respect to the probabilities themselves. This makes RCS the first proposal for quantum supremacy with both of these properties. Finally, we also give a natural condition under which an existing statistical measure, cross-entropy, verifies RCS, as well as describe a new verification measure which in some formal sense maximizes the information gained from experimental samples.

[*] Corresponding Author.

Establishing the exponential advantage of quantum computers over their classical counterparts was a crucial development in launching the field of quantum computation. The first evidence came in the form of complexity theoretic proofs that certain computational problems (of no practical value) can be solved exponentially faster by quantum computers in the black box model [1, 2], thus calling into question the Extended-Church Turing thesis, a foundational principle of classical complexity theory. Soon thereafter Shor's quantum factoring algorithm [3] provided a practically useful quantum speedup while at the same time giving a different type of evidence for the power of quantum computers — integer factorization is arguably the most well studied algorithmic problem — studied by number theorists going back to Fermat, and with particularly intense algorithmic efforts motivated by cryptography, including the RSA challenge.

With the recent progress in experimental realization of "noisy intermediate-scale" quantum computers (NISQ) [4, 5, 6, 7], the field is again at the threshold of a key milestone: *quantum supremacy*, i.e., the experimental realization of a computational task that cannot be solved in a reasonable amount of time by any classical means. As in the earliest demonstrations of "theoretical quantum supremacy", there is no requirement that the computational task be useful. The new challenge is that the computational task be experimentally realizable for near-term devices, thus ruling out standard computational tasks such as large-scale factoring which NISQ devices will not be capable of performing. Instead, all proposals for quantum supremacy have focused on sampling problems (e.g., [8, 9]), since the raw output of a quantum computer is a sample from a probability distribution resulting from a measurement. This choice, however, has important ramifications for the challenge of establishing computational difficulty of the task for any classical computer — both in the types of complexity theoretic techniques available for proving hardness and the relative lack of experience with the algorithmic difficulty of specific sampling problems.

Broadly speaking, we can classify supremacy proposals into two categories – those seeking to provide very strong complexity-theoretic evidence of classical intractability while hoping to be physically realized in the near term, versus those with excellent prospects for physical realization in the short term while providing weaker evidence of classical intractability. This paper shows that these categories intersect by providing strong complexity-theoretic evidence of classical intractability for the leading candidate from the latter category.

More specifically, the first category of quantum supremacy proposals had their origins in the desire to obtain strong complexity-theoretic evidence of the power of quantum computers. A key insight was that focusing on the probability distributions quantum devices can sample from, rather than more standard notions of computing or optimizing functions, opens up the possibility of strong evidence of classical intractability. This perspective led to proposals such as BosonSampling [8] and IQP [10], together with proofs that the probabilities of particular quantum outcomes correspond to quantities which are difficult to compute. This allowed them to connect the hardness of classical simulation of such systems to well-supported hardness assumptions stemming from complexity theory.

As an added bonus, Aaronson and Arkhipov realized that BosonSampling might be experimentally feasible in the near term, and helped jump-start the quest for quantum supremacy more than half a decade ago [11, 12, 13, 14]. More recently, BosonSampling experiments have faced experimental difficulties with photon generation and detector efficiency, making it challenging to push these experiments to the scale required to achieve supremacy (~ 50 photons) [15, 16]. It remains to be seen if such experiments can be implemented in the near future.

The second category of supremacy results is directly inspired by the dramatic experimental progress in building high-quality superconducting qubits (e.g., [4, 9]). These groups defined the natural sampling task for their experimental context, which we call *Random Circuit Sampling* (RCS). The task is to take an (efficient) quantum circuit of a specific form, in which each gate is chosen randomly, and generate samples from its output distribution. This proposal promises to be more readily scaled to larger system sizes in the near term. In particular, the group at Google/UCSB plans to conduct such an experiment on a 2D array of 49 qubits in the very near term [17]. However, RCS lacks some of the complexity-theoretic evidence that made BosonSampling so theoretically compelling – essentially because the quantum system is of a generic form which does not directly connect with complexity. To put this another way, usually the difficulty of simulating quantum algorithms comes from carefully engineered constructive and destructive interference patterns. However RCS by definition only reproduces "generic" interference patterns, and there is thus far no evidence that these are difficult to reproduce classically.

Our main result gives strong complexity-theoretic support to this experimentally driven proposal. In particular, we rely on a characterization of the output distribution of quantum circuits using Feynman path integrals as a stepping stone to showing that computing output probabilities of random quantum circuits is computationally hard. These tools are directly relevant to the upcoming superconducting experiment of Google/UCSB but will be useful to understand the capabilities of many other near-term experiments. Taken in combination with recent results establishing a subsequent piece of evidence for hardness for such systems [18, 19], our result puts RCS on par with the strongest theoretical proposals for supremacy including BosonSampling.

There is one more ingredient of a quantum supremacy proposal such as RCS, namely verifying that an experimental realization of Random Circuit Sampling has performed RCS faithfully. The two leading proposals for verification, cross-entropy and Heavy Output Generation (HOG), are only known to work under strong, and distinct assumptions. Cross-entropy verifies supremacy under a very strong assumption about the error model, and HOG verifies supremacy under a strong complexity assumption. Here we show how to greatly relax the assumptions under which cross-entropy verifies supremacy. In particular, we show that if the entropy of the device's output distribution is greater than the ideal entropy, then cross-entropy verifies supremacy, through our complexity arguments. This condition would follow assuming some natural local noise models.

It turns out that, viewed from the correct perspective, cross-entropy and HOG are more similar than it appears at first sight. This perspective allows us to formulate a new verification measure – Binned Output Generation (BOG), a common generalization of the two and has the property that it works if either does. In addition, it is the optimal verification measure in a certain formal sense.

1 Average-case hardness

Proposals for quantum supremacy have a common framework. The computational task is to sample from the output distribution D of some experimentally feasible quantum process or algorithm (on some given input). To establish quantum supremacy we must show

1. *Hardness*: no efficient classical algorithm can sample from any distribution close to D, and

2. *Verification*: an algorithm can check that the experimental device sampled from an output distribution close to D.

This need for verifiability effectively imposes a robustness condition on the difficulty of sampling from D. For example, the ability to sample one particular output x of a quantum circuit with the correct probability $D(x)$ is known to be hard for classical computers, under standard complexity assumptions, e.g. [10, 20, 21, 22, 23]. But this is not a convincing proof of supremacy – for one, under any reasonable noise model, this single output probability $D(x)$ might not be preserved. Moreover, this single output x is exponentially unlikely to occur – and would therefore be extremely difficult to verify. Accordingly, any convincing proof of quantum supremacy must establish that D is actually uniformly difficult to sample from. That is, the computational difficulty must be embedded across the entire distribution, rather than concentrated in a single output.

The starting point for the BosonSampling proposal of Aaronson and Arkhipov consisted of three observations: (1) In general, for sufficiently hard problems (think #P-hard), showing a distribution D is uniformly difficult to sample from corresponds to showing that for most outputs x, it is hard to compute $D(x)$. In complexity theory, this is referred to as "average-case" rather than "worst-case" hardness. (2) The output probabilities of systems of noninteracting bosons can be expressed as permanents of certain $n \times n$ matrices. (3) By a celebrated result of Lipton [24], computing permanents of random matrices is #P-hard, or truly intractable in the complexity theory pantheon. Together, these gave convincing evidence of the hardness of sampling typical outputs of a suitable system of noninteracting bosons, which could be experimentally feasible in the near term.

Specifically, they proved that no classical computer can sample from any distribution within inverse-exponential total variation distance of the ideal BosonSampling output distribution. Formally extending these results to experimentally relevant noise models, such as constant total variation distance, seems to require approximation robust average-case hardness that is beyond the reach of current methods. Nevertheless, their average-case hardness results are important as they establish a necessary foundation for noise-tolerant quantum supremacy of BosonSampling.

Permanents have a special structure enabling their average-case hardness – an ingredient which is thus far missing in other supremacy proposals. Technically, average-case hardness is established by creating a "worst-to-average-case reduction". We will show such a reduction for RCS. At a high level, such reductions are based on error-correcting codes, which are becoming more prominent across diverse areas of physics (see e.g., [25]); just as an error-correcting code allows one to recover an encoded message under the corruption of some of its entries, a worst-to-average-case reduction allows one to recover a worst-case solution from an algorithm that works most of the time. More formally, such reductions involve showing that the value of a worst-case instance x can be efficiently inferred from the values of a few random instances $r_1, \dots, r_m$. For this to be possible at all, while the r_k might be individually random, their correlations must depend upon x (which we shall denote by r_0). Typically such reductions rely on a deep global structure of the problem, which makes it possible to write the value at r_k as a polynomial in k of degree at most m. For example, the average-case property of permanents is enabled by its algebraic structure – the permanent of an $n \times n$ matrix can be expressed as a low degree polynomial in its entries. This allows the value at $r_0 = x$ to be computed from the values at r_k by polynomial interpolation.

An astute reader may have noticed that randomizing the instance of permanent corresponds to starting with a random linear-optical network for the BosonSampling experiment, but still focusing on a fixed output. Our goal however was to show for a fixed experiment that it is hard to calculate the probability of a random output. These are equivalent by a technique called "hiding". By the same token, it suffices for RCS to show that it is hard to compute the probability of a fixed output, 0, for a random circuit C.

To show this average-case hardness for quantum circuits, we start with the observation that the probability with which a quantum circuit outputs a fixed outcome x can be expressed as a low degree multivariate polynomial in the parameters describing the gates of the circuit, thanks to writing the acceptance probability as a Feynman path integral. Unfortunately, this polynomial representation of the output probability does not immediately yield a worst-to-average-case reduction. At its core, the difficulty is that we are not looking for structure in an individual instance – such as the polynomial description of the output probability for a particular circuit above. Rather, we would like to say that several instances of the problem are connected in some way, specifically by showing that the outputs of several different related circuits are described by a low degree (univariate) polynomial. With permanents, this connection is established using the linear structure of matrices, but quantum circuits do not have a linear structure, i.e. if A and B are unitary matrices, then $A + B$ is not necessarily unitary. This limitation means one cannot directly adapt the proof of average-case hardness for the permanent to make use of the Feynman path integral polynomial.

Here is a more sophisticated attempt to connect up the outputs of different circuits with a polynomial: Suppose we take a worst-case circuit $G = G_m \dots G_1$, and multiply each gate G_j by a Haar-random matrix H_j. By the invariance of the Haar measure, this is another random circuit – it is now totally scrambled. Now we invoke a unique feature of quantum computation, which is that it is possible to implement a fraction of a quantum gate. This allows us to replace each gate H_j with $H_j e^{\ \ i\theta h_j}$, where $h_j = \ \ i \log H_j$ and θ is a small angle, resulting in a new circuit $G(\theta)$. If $\theta = 1$ this gives us back the worst-case circuit $G(1) = G$, but if θ is very tiny the resulting circuit looks almost uniformly random. One might now hope to interpolate the value of $G(1)$ from the values of $G(\theta_k)$ for many small values of θ_k, thus effecting a worst-to-average case reduction. Unfortunately, this doesn't work either. The problem is that $e^{\ \ i\theta h_j}$ is not a low degree polynomial in θ, and therefore neither is $G(\theta)$, so we have nothing to interpolate with.

The third attempt, which works, is to consider using a truncated Taylor series of $e^{\ \ i\theta h_j}$ in place of $e^{\ \ i\theta h_j}$ in the above construction. The values of the probabilities in this truncation will be very close to those of the proposal above – and yet by construction we have ensured our output probabilities are low degree polynomials in theta. Therefore, if we could compute most output probabilities of these "truncated Taylor" relaxations of random circuits, then we could compute a worst-case probability.

Theorem 1 (Simplified) *It is* $\#\mathsf{P}$*-hard to exactly compute* $|\langle 0|C'|0\rangle|^2$ *with probability* $\frac{3}{4}+\frac{1}{\mathsf{poly}(n)}$ *over the choice of* C'*, where each gate of* C' *is drawn from any one of a family of discretizations of the Haar measure.*

These truncated circuit probabilities are slightly different from the average-case circuit probabilities but are exponentially close to them (even in relative terms). However, they are essentially the same from the perspective of supremacy arguments because quantum supremacy requires that computing most output probabilities even approximately remains #P-hard, and our perturbations to the random circuits fall within this approximation window. Therefore, we have established a form of worst-to average-case reduction which is necessary, but not sufficient, for the supremacy condition to remain true. This is directly analogous to the case of permanents, where we know that computing average-case permanents exactly is #P-hard, but we do not know this reduction is sufficiently robust to achieve quantum supremacy. For more details, see the Methods (Section 3.1).

RCS does satisfy an additional robustness property known as "anti-concentration". Anti-concentration states that the output distribution of a random quantum circuit is "spread out" – that most output probabilities are reasonably large. Therefore, any approximation errors in estimating these probabilities are small relative to the size of the probability being computed. Once one has established a worst-to-average-case reduction, anti-concentration implies that there is some hope for making this reduction robust to noise – intuitively it says that the signal is large compared to the noise.

Of the numerous quantum supremacy proposals to date which are conjectured to be robust to noise [8, 9, 19, 26, 27, 28, 29, 30, 31, 32], only two have known worst-to-average-case reductions: BosonSampling and FourierSampling [8, 26]. However, it remains open if these proposals also anti-concentrate. On the other hand, many supremacy proposals have known anti-concentration theorems (see e.g., [9, 19, 27, 29, 30, 31, 32]), but lack worst-to-average-case reductions. We note, however, that anti-concentration is arguably less important than worst-to-average case reductions, as the latter is necessary for quantum supremacy arguments, while the former is not expected to be necessary. In the case of RCS, anti-concentration follows from prior work [18, 19]. Therefore, our work is the first to show that both can be achieved simultaneously.

The leading quantum supremacy proposals

Proposal	*Worst-case hardness*	*Exact average-case hardness*	*Approximate average-case hardness*	*Anti-concentration*	*Feasible experiment?*
BosonSampling[a]	✓	✓			
FourierSampling[b]	✓	✓			
IQP[c]	✓			✓	
Random Circuit Sampling[d]	✓	$\textcircled{\checkmark}$		✓	✓

[a] [8]. [b] [26]. [c] [10, 27, 29]. [d] [9, 19, 33, 34].

Table 1: Here we list the leading quantum supremacy proposals and summarize their known complexity-theoretic properties.

2 Statistical verification of Random Circuit Sampling

We now turn to verifying that an experimental realization of Random Circuit Sampling has performed RCS faithfully. Verification turns out to be quite challenging. The first difficulty is that computing individual output probabilities of an ideal quantum circuit requires exponential classical time. However, current proposals leverage the fact that near-term devices with around $n = 50$ qubits are small enough that it is feasible to perform this task on a classical supercomputer (inefficiently), but large enough that the quantum devices solves an impressively difficult problem. While this might seem contradictory to the claim of quantum supremacy, note that the task which is (barely) feasible with an inefficient algorithm is only to compute individual probabilities. In contrast, naïvely simulating the sampling experiment would require far more – either computing all the probabilities or computing a smaller number of conditional probabilities. The second difficulty is that one can only take a small number of samples from the experimental quantum device. This means there is no hope of experimentally observing all 2^{50} outcomes, nor of estimating their probabilities empirically. The challenge is therefore to develop a statistical measure which respects these limitations, and nevertheless verifies quantum supremacy.

A leading statistical measure proposed for verification is the "cross-entropy" measure [9, 35, 34], which, for a pair of distributions D and D' is defined as:

$$\mathsf{CE}(D, D') = \sum_{x \in \{0,1\}^n} \left(D(x) \log \left(\left(\frac{1}{D'(x)} \right) \right(\right($$

For RCS it is being used as a measure of the distance between the output distribution of the experimental device tuned to perform the unitary U, denoted p_{dev}, and the ideal output distribution of the random circuit under U, denoted p_U.

A useful feature of this measure is that it can be estimated by taking a few samples x_i from the device and computing the average value of $\log(1/p_U(x_i))$ using a classical supercomputer. By concentration of measure arguments this converges very quickly to the true value.

Ideally, we would like to connect the cross-entropy measure to the rigorous complexity-theoretic arguments in favor of quantum supremacy developed in Section 1, which require closeness in total variation distance to the ideal. Without any assumptions as to how the device operates, it is easy to see that cross-entropy cannot verify total variation distance directly, as the latter requires exponentially many samples to verify.

However, we show that there is a natural assumption under which the cross-entropy measure certifies closeness in total variation distance. Namely, if one assumes that the entropy of the experimental device is greater than the entropy of the ideal device, then scoring well in cross-entropy does certify closeness in total variation distance:

Claim 2 *If $H(p_{dev}) \quad H(p_U)$, then achieving a cross-entropy score which is ϵ-close to ideal, i.e., $|\mathsf{CE}(p_{dev}, p_U) \quad H(p_U)| \leq \epsilon$, implies that $\|p_{dev} \quad p_U\| \leq \sqrt{\epsilon/2}$.*

The proof of this fact follows from Pinsker's inequality. A similar observation was recently independently obtained by Brandão (in personal communication). This assumption would follow from a number of natural noise models – such as local depolarizing noise, but not others – such as forms of erasure. Therefore, to understand the utility of cross-entropy it is crucial to characterize the noise present in near-term devices. We also use this as intuition to construct distributions which score well on cross-entropy but are far in total variation distance – we start with the ideal distribution and lower entropy.

A concurrent proposal of Aaronson and Chen, known as "Heavy Output Generation" or HOG, studied a different avenue to supremacy. Aaronson and Chen conjectured that given a randomly chosen quantum circuit C, it is difficult to output strings which have "above median" mass in C's output distribution. This proposal connects directly to a statistical test for verification, and the hardness of this task was connected to a non-standard complexity-theoretic conjecture known as QUATH.

To generalize these verification proposals, we describe a new statistical measure which we call BOG ("Binned Output Generation") which is a common ancestor to both cross-entropy and HOG and yet which is still easy to estimate from experimental data. In particular, this means that BOG verifies supremacy if either the entropy assumption or QUATH holds. Indeed viewed from the right perspective, these measures are more similar than it appears at first sight. In some formal sense BOG maximizes the amount of information one gains in the course of computing HOG or cross-entropy, and is therefore the optimal verification measure if one can only take polynomially many samples from the experimental device. For more details, see the Methods (Section 3.2).

3 Methods

3.1 Worst-to-average-case reduction

Our first result gives evidence that approximating average-case output probabilities of random quantum circuits remains difficult. It is well known that computing output probabilities of worst-case quantum circuits is #P-hard. Our goal is, therefore, to establish that computing output probabilities of *average-case* random quantum circuits is just as difficult. We achieve this by giving a *worst-to-average-case reduction* for computing output probabilities of random quantum circuits. That is, we show that if one could compute average-case quantum circuit probabilities, then one could infer the value of worst-case quantum circuit probabilities. Therefore, computing average-case probabilities is also #P-hard.

Establishing average-case hardness is surprisingly subtle. It will be useful to first recall the worst-to-average-case reduction for the permanent of matrices over the finite field $\mathbb{F}_q$ [24], where q is taken to be a sufficiently large polynomial in the input parameter. In the case of permanents, the structure which connects the values of random permanents is low-degree polynomials. The permanent of an $n \times n$ matrix,

$$\mathsf{perm}(A) = \sum_{\in S_n} \prod_{i=1}^{n} \Bigg(A_{i, \ (i)}$$

is a polynomial of degree n in the n^2 matrix entries. Let X be a random $n \times n$ matrix over $\mathbb{F}_q$, where $q \quad n+2$. Moreover, suppose our goal is compute the permanent of a worst-case matrix Y. We first consider the line $A(t) = Xt + Y$; note that for $t \neq 0$, $A(t)$ is uniformly distributed over $\mathbb{F}_q^{n\times n}$. If we are able to calculate $\mathsf{perm}(R)$ with probability $\quad 1 \quad \frac{1}{3(n+1)}$ over $R \sim_{\mathcal{U}} \mathbb{F}_q^{n\times n}$, then by the union bound, we could compute $A(t)$ correctly at $n+1$ different values of t with bounded error probability. This is possible because the union bound holds despite $A(t)$ being correlated with one another – it only requires that the *marginal* distribution on each one is uniform. So standard polynomial interpolation techniques on $\{(t_j, \mathsf{perm}(A(t_j))\}_{j=1,\ldots,n+1}$ allow us to learn the function $\mathsf{perm}(A(t))$ and therefore, $\mathsf{perm}(Y) = \mathsf{perm}(A(0))$. With more rigorous analysis – but the same intuition – one can argue that we only need to be calculate $\mathsf{perm}(R)$ with probability $3/4 + 1/\mathsf{poly}(n)$ [36, 37].

Therefore, polynomial interpolation allows us to compute permanents of every matrix $\in \mathbb{F}_q^{n\times n}$ if we can compute the permanent on most matrices. A "random survey" of permanent values can be used to infer the value of all permanents. Combined with the fact that the permanent problem is worst-case #P-hard [38], this implies that computing permanents in $\mathbb{F}_q^{n\times n}$ on average is #P-hard. Formally, the following result was obtained.

Theorem 3 (Average-case hardness for permanents [24, 37]) *The following is #*P*-hard: For sufficiently large q, given a uniformly random matrix $M \in \mathbb{F}_q^{n\times n}$, output* $\mathsf{perm}(M)$ *with probability* $\quad \frac{3}{4} + \frac{1}{\mathsf{poly}(n)}$.

To establish worst-to-average-case reductions for random circuits, we need to find a similar structural relation between the circuit whose output probability we wish to compute, and average-case circuits in which each gate is chosen randomly. A first observation is that there is indeed a low-degree polynomial structure – stemming from the Feynman path-integral – which allows us to write the probability of any outcome as a low-degree polynomial in the gate entries. This polynomial is fixed once we fix both the outcome and the architecture of the circuit, and the degree is twice the number of gates in the circuit (where the factor of 2 accounts for the Born rule for output probabilities) which is a polynomial in the input parameter.

Fact 4 (Feynman path integral) *Let $C = C_m C_{m \quad 1} \ldots C_2 C_1$, be a circuit formed by individual gates C_i acting on n qubits. Then*

$$\langle y_m | C | y_0 \rangle = \sum_{y_1, y_2, \ldots, y_{m-1} \in \{0,1\}^n} \prod_{j=1}^{m} \Bigg(\langle y_j | C_j | y_{j \quad 1} \rangle .$$

There are two subtleties we need to address. The first is that the value of this polynomial is the probability of a fixed output y_m. Our analysis will therefore focus on the hardness of estimating the

$$\mathsf{p_0}(C) \overset{\mathsf{def}}{=} |\langle 0^n|C|0^n\rangle|^2$$

probability for C drawn from $\mathcal{H}_{\mathcal{A}}$, rather than the hardness of approximating the probability of a random y_m. These can be proven equivalent using the "hiding" property of the $\mathcal{H}_{\mathcal{A}}$ distribution: we can take a circuit drawn from this distribution, append Pauli X gates to a uniformly chosen subset of output qubits, and remain distributed via $\mathcal{H}_{\mathcal{A}}$. We discuss hiding in more detail in Section 1.5 of the Supplementary Information.

The second subtlety is that this is a polynomial over the complex numbers, instead of $\mathbb{F}_q$. Bridging this gap requires considerable technical work. We note that Aaronson and Arkhipov have given a worst-to-average-case reduction for computing the permanent with complex Gaussian entries [8]. However, our reduction will be quite different, due to structural differences between quantum circuit amplitudes and permanents. Indeed, in proving the reduction for permanents of matrices over finite fields, we leveraged the fact that $A(t) = Xt + Y$ will be randomly distributed across $\mathbb{F}_q^{n\times n}$ since X is uniformly random and Y is fixed. To leverage a similar property for random circuit sampling, we need, given a (possibly worst-case) circuit C, a polynomial $C(t)$ over circuits such that (1) $C(0) = C$ and (2) $C(t)$ is distributed like a $\mathcal{H}_{\mathcal{A}}$. To be more precise, for a fixed architecture $\mathcal{A}$, we will we hope to say that the $\mathsf{p_0}(C)$ probability for a circuit $C \sim \mathcal{H}_{\mathcal{A}}$ is hard to compute on average.

A naïve approach to doing this is to copy the proof for the permanent. If we could perturb each gate in a random linear direction, then we could use polynomial interpolation to perform the worst-to-average-case reduction as above. That is, consider taking a worst-case circuit A and adding a random circuit B (gate by gate) to obtain $A + tB$. It is true that $\mathsf{p_0}(A + tB)$ is a low-degree polynomial in t, so one might hope to use interpolation to compute the worst-case value at $t = 0$. Unfortunately, this idea does not work because quantum gates do not have a linear structure. In other words, if A and B are unitary matrices, then $A+tB$ is not necessarily unitary – and hence $A + tB$ are not necessarily valid quantum circuits. So this naïve interpolation strategy will not work.

We consider a different way of perturbing circuits, which makes use of the unique properties of quantum mechanics. Suppose that we take a (possibly worst-case) circuit $C = C_m, \ldots, C_1$, and multiply each gate C_j by an independent Haar random matrix H_j. That is, we replace each gate C_j with C_jH_j. By the left-invariance of the Haar measure, this is equivalent to selecting each gate uniformly at random – i.e. it is equivalent to $\mathcal{H}_{\mathcal{A}}$. We have now recovered our original distribution over circuits, but in some sense we have gone too far, as we have completely erased all of the information of our worst-case circuit C. To remedy this, we will make use of a uniquely quantum ability – namely, that it is possible to perform a fraction of a quantum gate. This has no classical analog (indeed, what would it mean to perform 1/10 of a NAND gate?) That is, suppose we "rotate back" by tiny amount back towards C_j by some small angle θ. More specifically, replace each gate C_j of the circuit with $C_jH_je^{\ \ ih_j\theta}$ where $h_j = \ \ i\log H_j$. If $\theta = 1$ this gives us back the circuit C, but if θ is very tiny then each gate looks almost Haar random. One might hope that by collecting the values of many probabilities at small angles θ, one could interpolate back to the point C of interest. Therefore, a second attempt would be to take the circuit C, scramble it by multiplying it gate-wise by a *perturbed Haar distribution* defined below, and then use some form of interpolation in θ to recover the probability for C at $\theta = 1$.

Definition 5 (θ-perturbed Haar-distribution) *Let $\mathcal{A}$ be an architecture over circuits, θ a constant $\in [0, 1]$, and let $G_m, \ldots, G_1$ be the gate entries in the architecture. Define the distribution $\mathcal{H}_{\mathcal{A},\theta}$ over circuits in $\mathcal{A}$ by setting each gate $G_j = H_je^{\ \ ih_j\theta}$ where H_j is an independent Haar random unitary and $h_j = \ \ i\log H_j$.*

Unfortunately, this trick is not sufficient to enable the reduction. The problem is that $e^{\ \ i\theta h_j}$ is not a low-degree polynomial in θ, so we have no structure to apply polynomial interpolation onto. While there is structure, we cannot harness it for interpolation using currently known techniques. Although this does not work, this trick has allowed us to make some progress. A promising property of this method of scrambling is that it produces circuits which are close to randomly distributed – which we will later prove rigorously. This is analogous to the fact that $A + tB$ is randomly distributed in the permanent case, a key property used in that proof. We merely need to find some additional polynomial structure here in order to utilize this property.

We find this polynomial structure by considering Taylor approximations of $e^{\ \ ih_j\theta}$ in place of $e^{\ \ ih_j\theta}$ in the above construction. While these truncated circuits are slightly non-unitary, the values of the probabilities in this truncation will be very close to those of the proposal above – and yet by construction we have ensured our output probabilities are low degree polynomials in θ. Formally, we define a new distribution over circuits with this property:

Definition 6 ((θ, K)-truncated perturbed Haar-distribution) *Let $\mathcal{A}$ be an architecture over circuits, θ a constant $\in [0,1]$, K an integer, and let $G_m, \ldots, G_1$ be the gate entries in the architecture. Define the distribution $\mathcal{H}_{\mathcal{A},\theta,K}$ over circuits in $\mathcal{A}$ by setting each gate*

$$G_j = H_j \quad \sum_{k=0}^{K} \frac{(\ \ ih_j\theta)^k}{k!}\Bigg)$$

where H_i is an independent Haar random unitary and $h_j = \ \ i \log H_j$.

Now suppose we take our circuit C of interest and multiply it by $\mathcal{H}_{\mathcal{A},\theta,K}$ gate-by-gate to "scramble" it. This is precisely how a classical computer would sample from $C \cdot \mathcal{H}_{\mathcal{A},\theta}$ (where the multiplication is performed gatewise) as one cannot exactly represent a continuous quantity digitally. Suppose we could compute the probabilities of these circuits for many choices of θ with high probability. Now one can use similar polynomial interpolation ideas to show hardness of this task.

To state this formally, let us define some notation. For a circuit C and $\mathcal{D}$ a distribution over circuits of the same architecture, let $C \cdot \mathcal{D}$ be the distribution over circuits generated by sampling a circuit $C' \sim \mathcal{D}$ and outputting the circuit $C \cdot C'$ (where again, the multiplication is performed gatewise). Explicitly, we show the following worst-to-average-case reduction.

Theorem 1 *Let $\mathcal{A}$ be an architecture so that computing $\mathsf{p_0}(C)$ to within additive precision $2^{\ \ \mathsf{poly}(n)}$, for any C over $\mathcal{A}$ is #P-hard in the worst case. Then it is #P-hard to exactly compute $\frac{3}{4} + \frac{1}{\mathsf{poly}(n)}$ of the probabilities $\mathsf{p_0}(C')$ over the choice of C' from the distributions $\mathcal{D}'_C \overset{\mathsf{def}}{=} C \cdot \mathcal{H}_{\mathcal{A},\theta,K}$ where $\theta = 1/\mathsf{poly}(n)$, $K = \mathsf{poly}(n)$.*

A formal proof of Theorem 1, as well as commentary on its relation to the hardness conjectures needed to establish quantum supremacy, are provided in the Supplementary Information. For example, although we have changed the distribution over which average-case hardness is extablished, we show that hardness over the new distribution, Theorem 1, is *necessary* for the average-case conjecture relevant to the quantum supremacy of RCS to be true. For details, see Section 1.2 of the Supplementary Information.

3.2 Verification of Random Circuit Sampling

In this section we discuss the verification of Random Circuit Sampling experiments. Let us first recall the setting: we are given a description of a randomly generated quantum circuit (which we will refer to as the ideal circuit) as well as an experimental device that outputs samples.

We wish to verify that no efficient classical device could have produced this output. One difficulty which immediately arises is that one can only take a small (polynomial) number of samples from the device, and therefore one cannot characterize the entire output distribution of the device. Another basic difficulty is that the output of the ideal circuit is computationally hard to produce — so for large system sizes we do not even know what to compare the output of the experimental device to. Current verification schemes leverage the fact that intermediate size experiments, such as $n = 50$ qubit systems, are small enough that it is feasible to calculate on a classical supercomputer the probability p_x that the ideal quantum circuit outputs a particular string x. The list of strings x output by the device, together with p_x, is summarized in a statistical score which can efficiently estimated with few samples from the device. Our goal is to understand under what circumstances such a score verifies quantum supremacy.

In this section, if unspecified, a probability distribution will be over strings $x \in \{0,1\}^n$. The size of the domain will be denoted $N = 2^n$. The phrase "with high probability" will mean with probability $1 \quad o(1)$.

3.2.1 The cross-entropy supremacy proposal

Cross-entropy is a leading proposal for verifying quantum supremacy [9, 34, 35]. Recall from Section 2 that the cross-entropy between distributions D and D' is defined as $\mathsf{CE}(D, D') = \sum_{x\in\{0,1\}^n} \Big(D(x) \log \Big(\Big(\frac{1}{D'(x)} \Big)$. For RCS it is being used as a measure of the distance between the output distribution of the experimental device tuned to perform the unitary U, denoted p_{dev}, and the ideal output distribution of the random circuit under U, denoted p_U [9, 34, 35]. Estimating $\mathsf{CE}(p_{dev}, p_U)$ requires taking merely $k \ll N$ samples, $x_1, \ldots, x_k$, from the experimental device, followed by the computation of the empirical estimate E of the cross-entropy

$$E = \frac{1}{k} \sum_{i=1\ldots k} \Big(\log \Big(\Big(\frac{1}{p_U(x_i)} \Big) \Big(\qquad (1)$$

by using a supercomputer to calculate ideal probabilities $p_U(x_i) = |\langle x_i|U|0^n\rangle|^2$ for only the observed outcome strings x_i. By concentration of measure, for typical U, after polynomially many samples, E will converge to $\mathsf{CE}(p_{dev}, p_U)$. This follows from the fact that with high probability over the choice of random unitary, the largest and smallest ideal outcome probability are of order $\log N/N$ and $1/N^2$, respectively. Hence the logarithms of all $p_U(x_i)$ are within a constant factor of one another (on average), so by the Chernoff bound one can estimate this quantity to multiplicative error ϵ with merely $\log(1/\epsilon)$ samples.

The goal of their experiment is to score as close to the ideal expectation value as possible (on average over the choice of U). In fact, this measure has become incredibly important to the Google/UCSB group: it is being used to calibrate their candidate experimental device [17, 34].

3.2.2 The relationship between cross-entropy and total variation distance

As before, let p_U be the ideal output distribution and p_{dev} be the output distribution of the experimental device. To motivate the cross-entropy score, the Google/UCSB paper assumes that p_{dev} is a convex combination of p_U with the uniform distribution [9]. Here we show that scoring well in cross-entropy certifies closeness in total-variation distance under a considerably weaker assumption, namely:

Assumption 7 $H[p_{dev}] \quad H[p_U]$.

Claim 8 *If Assumption 7 holds, then achieving a cross-entropy score which is ϵ-close to ideal, i.e., $|\mathsf{CE}(p_{dev}, p_U) \quad H(p_U)| \le \epsilon$, implies that $\|p_{dev} \quad p_U\| \le \sqrt{\epsilon/2}$.*

Proof: (*Sketch*)

The claim follows from a straightforward application of Pinkser's inequality:

$$\|p_{dev} \quad p_U\| \le \sqrt{\frac{\|p_{dev} \quad p_U\|_{KL}}{2}} \qquad (2)$$

$$= \sqrt{\frac{\mathsf{CE}(p_{dev}, p_U) \quad H(p_{dev})}{2}} \qquad (3)$$

$$\le \sqrt{\frac{\mathsf{CE}(p_{dev}, p_U) \quad H(p_U)}{2}} \le \sqrt{\frac{\epsilon}{2}} \qquad (4)$$

Where equation 2 is Pinsker's inequality, equation 3 follows from the definition of KL divergence $\|D, D'\|_{KL} = \mathsf{CE}(D, D') \quad H(D)$ and equation 4 follows from Assumption 7.

□

It might appear at first sight that Assumption 7 follows from the very reasonable physical assumption that the experimental device is any noisy version of the ideal device. While this is not true in general, it does hold for some standard noise models such as local depolarizing noise. So one approach to verification is to verify the noise model and show that it is consistent with Assumption 7. We note that of course, verifying Assumption 7 directly would require exponentially many samples from the device.

Following this connection further, one can easily construct examples of distributions which score well on cross-entropy but are far from ideal in total variation distance. In particular one can achieve this by taking the ideal distribution and reducing its entropy.

Theorem 9 *For every unitary U, there exists a distribution D_U such that, with probability $1 \quad o(1)$ over the choice of U from the Haar measure, $|D_U \quad p_U| \quad 0.99$, and yet $\mathsf{CE}(D_U, p_U)$ is $O(1/N^{\Theta(1)})$-close to ideal.*

We provide a proof of Theorem 9 in the Supplementary Information.

3.2.3 Binned output generation (BOG): a common ancestor to cross-entropy and HOG

Recall that the goal of these statistical measures is to verify RCS with very few samples from the experimental device, but allowing for exponential classical postprocessing time (See Section 2). These tests are all performed by taking $k = \mathsf{poly}(n)$ samples from the device $x_1 \ldots x_k$ and then computing statistics on the ideal output probabilities $p_U(x_i)$ of the observed strings. It is natural to try to maximize the amount of information obtained from the computed values of $p_U(x_i)$, so as to eliminate the largest number of imposter distributions. In this section, we describe a statistical measure which performs this task. It simultaneously generalizes both HOG and cross-entropy difference – that is, passing this test implies that one has scored well on both cross-entropy and HOG. Furthermore, this test eliminates more imposter distributions than naively combining cross-entropy and HOG. As discussed previously, cross-entropy and HOG certify supremacy under two very different assumptions – one relating to the noise present in the device, and another to a non-standard complexity conjecture. Therefore this new measure verifies quantum supremacy if either assumption holds. We call this measure "binned output generation" or BOG, which we define below.

Consider dividing the interval $[0, 1]$ into $\mathsf{poly}(n)$ bins, such that for each bin $[a/N, b/N]$, we have $\int_a^b qe^{\quad q} = \Theta(1/\mathsf{poly}(n))$. In other words when sampling from a Porter-Thomas distribution, one would expect to see roughly an equal number of counts of $p_U(x_i)$ in each bin. Now suppose that one takes $k = \mathsf{poly}(n)$ samples from an experimental device (with a randomly chosen U) to obtain strings $x_1 \ldots x_k$. We say that the test passes if one has approximately the correct frequency of counts of $p_U(x_i)$ in each bin (up to small constant multiplicative error). By concentration of measure the ideal distribution will pass the test with high probability.

BOG can be seen as a simple refinement of HOG, where we divide the output probabilities into $\mathsf{poly}(n)$ bins instead of two (below median and above median). Therefore, this measure both verifies HOG and additionally ensures that the more fine-grained "shape" of the distribution is present as well. Furthermore, one can show that for a suitable choice of parameters, passing BOG implies that one has achieved nearly the ideal cross-entropy as well – as the ideal cross-entropy is $O(n)$, an $(1 \pm 1/\mathsf{poly}(n))$ multiplicative approximation to the cross-entropy suffices to verify $o(1)$ closeness to the ideal cross-entropy difference. BOG extracts the maximum amount of information out of the computation of the $p_U(x_i)$, as long as one ignores the higher-order bits of the results. Differences between these higher order bits are not observable with merely $\mathsf{poly}(n)$ samples from the device. For instance, if one passes BOG, then one has certified the expectation value of any Lipschitz function of the ideal probabilities to error $O(1/\mathsf{poly})$. In this particular sense BOG is information theoretically optimal. We therefore propose BOG as a measure for verification, as it uses the same data as HOG and cross-entropy to obtain more information about the output distribution.

Acknowledgements

We thank Scott Aaronson, Dorit Aharonov, Fernando Brandão, Matthew Coudron, Abhinav Deshpande, Tom Gur, Zeph Landau, Nicholas Spooner, and Henry Yuen for helpful discussions. Authors AB, BF, CN, and UV were supported by ARO Grant W911NF-12-1-0541 and NSF Grant CCF-1410022 and a Vannevar Bush faculty fellowship. BF is supported in part by an Air Force Office of Scientific Research Young Investigator Program award number FA9550-18-1-0148. Portions of this paper are a contribution of NIST, an agency of the US government, and is not subject to US copyright.

Author Contributions

All authors contributed equally to this work; author ordering is alphabetical.

Data Availability

The data that support the plots within this paper and other findings of this study are available from the corresponding author upon reasonable request.

References

[1] Bernstein, E. & Vazirani, U. V. Quantum complexity theory. In Kosaraju, S. R., Johnson, D. S. & Aggarwal, A. (eds.) *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing, May 16-18, 1993, San Diego, CA, USA*, 11–20 (ACM, 1993). URL `http://doi.acm.org/10.1145/167088.167097`.

[2] Simon, D. R. On the power of quantum cryptography. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, 116–123 (IEEE Computer Society, 1994). URL `https://doi.org/10.1109/SFCS.1994.365701`.

[3] Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review* **41**, 303–332 (1999).

[4] Mohseni, M. *et al.* Commercialize quantum technologies in five years. *Nature* **543**, 171–174 (2017).

[5] Kandala, A. *et al.* Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **549**, 242 – 246 (2017). URL `http://dx.doi.org/10.1038/nature23879`.

[6] Zhang, J. *et al.* Observation of a many-body dynamical phase transition with a 53-qubit quantum simulator. *Nature* **551**, 601–604 (2017).

[7] Preskill, J. Quantum Computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018). URL `https://doi.org/10.22331/q-2018-08-06-79`.

[8] Aaronson, S. & Arkhipov, A. The computational complexity of linear optics. In *Proceedings of the forty-third annual ACM Symposium on Theory of Computing*, 333–342 (ACM, 2011).

[9] Boixo, S. *et al.* Characterizing quantum supremacy in near-term devices. *Nature Physics* **14**, 595–600 (2018). URL `https://doi.org/10.1038/s41567-018-0124-x`.

[10] Bremner, M. J., Jozsa, R. & Shepherd, D. J. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 459–472 (The Royal Society, 2010).

[11] Spring, J. B. *et al.* Boson sampling on a photonic chip. *Science* 798–801 (2012).

[12] Broome, M. A. *et al.* Photonic boson sampling in a tunable circuit. *Science* **339**, 794–798 (2013).

[13] Tillmann, M. *et al.* Experimental boson sampling. *Nature Photonics* **7**, 540–544 (2013).

[14] Crespi, A. *et al.* Integrated multimode interferometers with arbitrary designs for photonic boson sampling. *Nature Photonics* **7**, 545–549 (2013).

[15] Neville, A. *et al.* No imminent quantum supremacy by boson sampling. *Nature Physics* **13**, 1153–1157 (2017). URL `http://dx.doi.org/10.1038/nphys4270`.

[16] Clifford, P. & Clifford, R. The classical complexity of boson sampling. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, 146–155 (SIAM, 2018).

[17] Martinis, J. The quantum space race (2018). Plenary talk at Quantum Information Processing (QIP) 2018, Available at https://collegerama.tudelft.nl/Mediasite/Showcase/qip2018/Channel/qip-day3.

[18] Brandão, F. G. & Horodecki, M. Exponential quantum speed-ups are generic. *Quantum Information & Computation* **13**, 901–924 (2013).

[19] Hangleiter, D., Bermejo-Vega, J., Schwarz, M. & Eisert, J. Anticoncentration theorems for schemes showing a quantum speedup. *Quantum* **2**, 65 (2018).

[20] Terhal, B. M. & DiVincenzo, D. P. Adaptive quantum computation, constant depth quantum circuits and Arthur-Merlin games. *Quantum Information & Computation* **4**, 134–145 (2004).

[21] Morimae, T., Fujii, K. & Fitzsimons, J. F. Hardness of classically simulating the one-clean-qubit model. *Physical Review Letters* **112**, 130502 (2014).

[22] Farhi, E. & Harrow, A. W. Quantum supremacy through the quantum approximate optimization algorithm. *arXiv:1602.07674* (2016).

[23] Bouland, A., Mancinska, L. & Zhang, X. Complexity Classification of Two-Qubit Commuting Hamiltonians. In Raz, R. (ed.) *31st Conference on Computational Complexity (CCC 2016)*, vol. 50 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 28:1–28:33 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2016). URL `http://drops.dagstuhl.de/opus/volltexte/2016/5846`.

[24] Lipton, R. J. New directions in testing. *Distributed Computing and Cryptography* 191–202 (1991).

[25] Pastawski, F., Yoshida, B., Harlow, D. & Preskill, J. Holographic quantum error-correcting codes: Toy models for the bulk/boundary correspondence. *Journal of High Energy Physics* **2015**, 149 (2015).

[26] Fefferman, B. & Umans, C. On the power of quantum Fourier sampling. In *11th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2016, September 27-29, 2016, Berlin, Germany*, 1:1–1:19 (2016). URL `http://dx.doi.org/10.4230/LIPIcs.TQC.2016.1`.

[27] Bremner, M. J., Montanaro, A. & Shepherd, D. J. Average-case complexity versus approximate simulation of commuting quantum computations. *Physical Review Letters* **117**, 080501 (2016).

[28] Aaronson, S. & Chen, L. Complexity-theoretic foundations of quantum supremacy experiments. In O'Donnell, R. (ed.) *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, vol. 79 of *LIPIcs*, 22:1–22:67 (Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017). URL `https://doi.org/10.4230/LIPIcs.CCC.2017.22`.

[29] Bremner, M. J., Montanaro, A. & Shepherd, D. J. Achieving quantum supremacy with sparse and noisy commuting quantum computations. *Quantum* **1**, 8 (2017). URL `https://doi.org/10.22331/q-2017-04-25-8`.

[30] Morimae, T. Hardness of classically sampling the one-clean-qubit model with constant total variation distance error. *Physical Review A* **96**, 040302 (2017).

[31] Bouland, A., Fitzsimons, J. F. & Koh, D. E. Complexity Classification of Conjugated Clifford Circuits. In Servedio, R. A. (ed.) *33rd Computational Complexity Conference (CCC 2018)*, vol. 102 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 21:1–21:25 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2018). URL `http://drops.dagstuhl.de/opus/volltexte/2018/8867`.

[32] Mann, R. L. & Bremner, M. J. On the complexity of random quantum computations and the Jones polynomial. *arXiv:1711.00686* (2017).

[33] Harrow, A. W. & Low, R. A. Random quantum circuits are approximate 2-designs. *Communications in Mathematical Physics* **291**, 257–302 (2009). URL `https://doi.org/10.1007/s00220-009-0873-6`.

[34] Neill, C. *et al.* A blueprint for demonstrating quantum supremacy with superconducting qubits. *Science* **360**, 195–199 (2018).

[35] Boixo, S., Smelyanskiy, V. N. & Neven, H. Fourier analysis of sampling from noisy chaotic quantum circuits. *arXiv:1708.01875* (2017).

[36] Welch, L. & Berlekamp, E. Error correction for algebraic block codes (1986). URL `https://www.google.com/patents/US4633470`. US Patent 4,633,470.

[37] Gemmell, P., Lipton, R., Rubinfeld, R., Sudan, M. & Wigderson, A. Self-testing/correcting for polynomials and for approximate functions. In *Proceedings of the Twenty-third Annual ACM Symposium on Theory of Computing*, STOC '91, 33–42 (ACM, New York, NY, USA, 1991). URL `http://doi.acm.org/10.1145/103418.103429`.

[38] Valiant, L. The complexity of computing the permanent. *Theoretical Computer Science* **8**, 189 – 201 (1979). URL `http://www.sciencedirect.com/science/article/pii/0304397579900446`.

On the Complexity and Verification of Quantum Random Circuit Sampling: Supplementary Information

1 Average-case hardness

Our main result is to give the first worst-to-average-case reduction for computing the output probabilities of random quantum circuits. We will now describe why this result is critical to establishing quantum supremacy from Random Circuit Sampling (RCS).

Let us first define what we mean by RCS. Random Circuit Sampling is the process of picking a random (efficient) quantum circuit and then sampling from its output distribution. Formally, an *architecture* $\mathcal{A}$ is a collection of directed acyclic graphs, one for each integer n. Each graph consists of $m \leq \mathsf{poly}(n)$ vertices where each vertex v has $\deg_{\text{in}}(v) = \deg_{\text{out}}(v) \in \{1, 2\}$ except for specific vertices s and t which have $\deg_{\text{out}}(s) = \deg_{\text{in}}(t) = n$ and $\deg_{\text{out}}(t) = \deg_{\text{in}}(s) = 0$. A circuit C acting on n qubits over $\mathcal{A}$ is instantiated by taking the n-th graph and specifying a gate for each vertex $v \notin \{s, t\}$ in the graph that acts on the qubits labelled by the edges adjacent to that vertex. The vertices s and t represent the state prior to and after the application of the circuit. That is, we can think of an architecture as an outline of a quantum circuit (one for each size n), and one needs to fill in the blanks (specify each gate) to instantiate a circuit.

We will consider the distribution on circuits where each gate is drawn uniformly at random. Here "uniformly at random" means according to the Haar measure, i.e. the unique measure on unitary matrices that is invariant under (left or right) multiplication by any unitary.

Definition 10 (Haar random circuit distribution) *Let $\mathcal{A}$ be an architecture over circuits and let the gates in the architecture be $\{G_i\}_{i=1,\ldots,m}$. Define the distribution $\mathcal{H}_{\mathcal{A}}$ over circuits in $\mathcal{A}$ by drawing each gate G_i independently from the Haar measure.*

Random Circuit Sampling is then defined as follows:

Definition 11 (Random Circuit Sampling) *Random Circuit Sampling over a fixed architecture $\mathcal{A}$ is the following task: given a description of a random circuit C from $\mathcal{H}_{\mathcal{A}}$, and a description of error parameters $\epsilon, \quad > 0$, with probability $1 \qquad$ over the choice of C, sample from the probability distribution induced by C (i.e., draw $y \in \{0,1\}^n$ with probability $\Pr(y) = |\langle y|C|0^n\rangle|^2$) up to total variation distance ϵ in time $\mathsf{poly}(n, 1/\epsilon)$. Here and throughout this paper, the probability distribution induced by C is $\Pr(y) = |\langle y|C|0^n\rangle|^2$ for $y \in \{0,1\}^n$.*

While RCS is defined relative to an architecture $\mathcal{A}$, the exact choice of $\mathcal{A}$ will not matter for our main result. In fact our result will still hold if some of the gates of $\mathcal{A}$ are fixed while others are drawn Haar-randomly. We discuss the architectures proposed for quantum supremacy in detail in Section 1.6 of the Supplementary Information. Also, note that this definition is designed to allow for a small amount of error in the classical sampler. This is to capture the fact that real-world quantum devices will be unable to perform this task exactly due to noise - and hence this definition allows the classical device the same error tolerance we allow the quantum device. As usual *total variation distance* means one half of the ℓ_1 distance between the probability distributions. Likewise, this definition allows the sampler to fail on a small fraction of inputs C; this is to ensure that randomly testing the quantum device for $\mathsf{poly}(n)$ random choices of C

suffices to certify that RCS has been performed correctly – otherwise one would need to test *all C* to verify the quantum device works.

The goal of our work is to argue that RCS is difficult for classical computers. The crux of this argument lies in the relative difference in the difficulty of estimating the output probabilities of classical vs quantum circuits. It is well known that certain output probabilities of quantum circuits are very difficult to compute – in fact, they can be #P-hard to approximate, which is truly intractable. In contrast, it is much easier to approximate the output probabilities of classical circuits [39], under an assumption known as the non-collapse of the polynomial hierarchy. This result alone is enough to establish the difficulty of *exactly* sampling from the probability distribution output by the quantum device (i.e. in the case $\epsilon = 0$) [8, 10].

However, to make this argument robust to experimental noise, we need the hardness of computing output probabilities to be "more spread out" in the output distribution, rather than concentrated in a single output which could be corrupted by noise. This was precisely the insight of Aaronson and Arkhipov [8]. They showed that BosonSampling cannot be classically simulated under the following conjecture:

Conjecture 12 ([8], Informal) *Approximating most output probabilities of most linear optical networks is #P-hard.*

While they did not prove this conjecture, they were able to prove the following necessary worst-to-average-case reduction:

Theorem 13 ([8], Informal) *Exactly computing most output probabilities of most linear optical networks is #P-hard.*

This result immediately implies that one cannot exactly sample from the output probability distributions of randomly chosen linear optical networks (i.e. $\epsilon = 0$ but $\neq 0$). Hence even "generic" optical interference patterns are difficult to generate classically; the remaining task is to make this average-to-worst case reduction robust to noise, and hence make the sampling result robust to experimental error ($\epsilon \neq 0$).

Following the arguments of Aaronson and Arkhipov, one can show that assuming the non-collapse of PH, no efficient classical algorithm can perform RCS, under the following approximate average-case hardness conjecture. We detail these arguments later in Section 1.5 of the Supplementary Information.

Conjecture 14 (Informal) *There exists an architecture $\mathcal{A}$ so that approximating $|\langle 0^n| C |0^n\rangle|^2$ for most $C \sim \mathcal{H}_{\mathcal{A}}$ is #P-hard.*

The astute reader may notice that we have stated Conjecture 14 in terms of a single output rather than a random output. These are equivalent by an argument known as "hiding" [8] which we detail in Section 1.5 of the Supplementary Information.

Our Theorem 1 establishes the analogue of Theorem 13 for Random Circuit Sampling. Just as for Aaronson and Arkhipov, this theorem will give necessary evidence in support of our main hardness conjecture:

Theorem 1(Simplified) *It is #P-hard to exactly compute $|\langle 0|C'|0\rangle|^2$ with probability $3/4 + 1/\mathsf{poly}(n)$ over the choice of C', where the circuit C' acting on n qubits is drawn from any one of a family of discretizations of $\mathcal{H}_{\mathcal{A}}$.*

This holds for any architecture $\mathcal{A}$ in which computing $|\langle 0|C'|0\rangle|^2$ is #P-hard in the worst case. Likewise this immediately implies one cannot exactly sample from "generic" interference patterns over systems of qubits. We further show that this average-case hardness theorem is in fact *necessary* for Conjecture 14 to be true in Section 1.3 of the Supplementary Information, and that an approximation-robust version of this theorem would be *sufficient* to prove Conjecture 14 in Section 1.4 of the Supplementary Information. We will provide further commentary on our result in Section 1.2 of the Supplementary Information after we prove it formally.

Furthermore, prior work has shown that Random Circuit Sampling has an additional property known as anti-concentration [18, 19], which has not been proven for BosonSampling or FourierSampling. Anti-concentration can be seen as evidence that an average-case hardness result could be made robust to noise. We will discuss how known anti-concentration results can be integrated into our hardness proof in Section 1.6 of the Supplementary Information.

1.1 Proof of Theorem 1

We will now prove Theorem 1, using the notation defined in Section 3.1 of the Methods.

Theorem 1 *Let $\mathcal{A}$ be an architecture so that computing $\mathsf{p_0}(C)$ to within additive precision $2^{\ \mathsf{poly}(n)}$, for any C over $\mathcal{A}$ is #P-hard in the worst case. Then it is #P-hard to exactly compute $\frac{3}{4} + \frac{1}{\mathsf{poly}(n)}$ of the probabilities $\mathsf{p_0}(C')$ over the choice of C' from the distributions $\mathcal{D}'_C \overset{\mathsf{def}}{=} C \times \mathcal{H}_{\mathcal{A},\theta,K}$ where $\theta = 1/\mathsf{poly}(n)$, $K = \mathsf{poly}(n)$.*

The proof of Theorem 1 follows by demonstrating the inherent *polynomial* structure of the problem and leveraging the structure via polynomial interpolation to equate average-case and worst-case hardness.

Proof: Let $\{H_j\}$ be a collection of independent Haar random gates and define

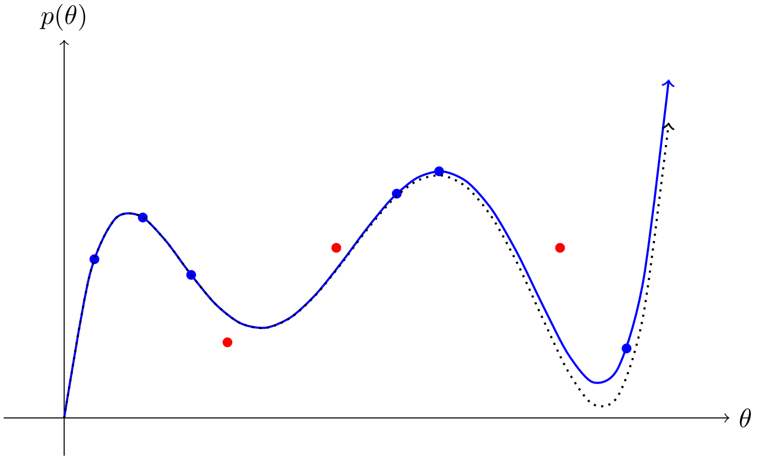
$$H'_j(\theta) = H_j \sum_{k=0}^{K} \frac{(\ \ i h_j \theta)^k}{k!}$$

where $h_j = \ \ i \log H_j$. Define the circuit $C'(\theta)$ as $C \times H'(\theta)$. Let $q(\theta) = \mathsf{p_0}(C'(\theta))$.

Notice that for a fixed choice of $\{H_j\}$, $q(\theta)$ is a low-degree polynomial in θ. By a Feynman path integral (Fact 4),

$$\langle y_m | C'(\theta) | y_0 \rangle = \sum_{y_1,\ldots,y_{m\ \ 1} \in \{0,\ldots,d\ \ 1\}^n} \prod_{j=1}^{m} \Big(\langle y_j | [C'(\theta)]_j | y_{j\ \ 1} \rangle$$

is a polynomial of degree mK as each term $\langle y_j | [C_1(\theta)]_j | y_{j\ \ 1} \rangle$ is a polynomial of degree K. Therefore, q is a polynomial of degree $2mK$. Now assume that there exists a machine $\mathcal{O}$ such that $\mathcal{O}$ can compute $\mathsf{p_0}(C')$ for $3/4 + 1/\mathsf{poly}(n)$ of C' where C' is drawn from the distribution in the theorem statement. A simple counting argument shows that for at least $\ 1/2 + 1/\mathsf{poly}(n)$ of the choices of $\{H_j\}$, $\mathcal{O}$ correctly computes $\mathsf{p_0}(C'(\theta))$ for at least $\ 1/2 + 1/\mathsf{poly}(n)$ of θ. Call such a choice of $\{H_j\}$ good.



Supplementary Figure 2: Example of a true function $\mathsf{p_0}(C)$ (dotted), inherent polynomial $q(\theta) = \mathsf{p_0}(C'(\theta))$ (solid), and potentially noisy samples $\{(\theta_\ell, \mathcal{O}(\theta_\ell))\}$.

Consider a machine $\mathcal{O}'$ with fixed $\theta_1, \ldots, \theta_k \in [0, \frac{1}{\mathsf{poly}(n)})$ that queries $\mathcal{O}(\theta_\ell)$ for $\ell = 1, \ldots, k$. Then $\mathcal{O}'$ applies the Berlekamp-Welch Algorithm [36] to compute a degree $2mK$ polynomial $\tilde{q}$ from the points $\{(\theta_\ell, \mathcal{O}(\theta_\ell))\}_{\ell=1,\ldots,k}$ and returns the output $\tilde{q}(1)$.

Theorem 15 (Berlekamp-Welch Algorithm [36]) *Let q be a degree d univariate polynomial over any field $\mathbb{F}$. Suppose we are given k pairs of $\mathbb{F}$ elements $\{(x_i, y_i)\}_{i=1,\ldots,k}$ with all x_i distinct with the promise that*

$y_i = q(x_i)$ for at least $\min(d+1, (k+d)/2)$ *points. Then, one can recover q exactly in* $\mathsf{poly}(k,d)$ *deterministic time.*

We remark that if we choose $k = 100mK$, then for a good $\{H_j\}$ with high probability (by a Markov's inequality argument), the polynomial $\tilde{q} = q$. Therefore, $\tilde{q}(1) = q(1) = \mathsf{p_0}(C'(1))$. Since at least $1/2 + 1/\mathsf{poly}(n)$ of $\{H_j\}$ are good, by repeating this procedure $\mathsf{poly}(n)$ times and applying a majority argument, we can compute $\mathsf{p_0}(C'(1))$ exactly. It only remains to show that $\mathsf{p_0}(C'(1))$ is a $2\ ^{\mathsf{poly}(n)}$ additive approximation to $\mathsf{p_0}(C)$, a #P-hard quantity.

Using standard bounds for Taylor series, one can easily show that $|\mathsf{p_0}(C'(1)) \quad \mathsf{p_0}(C)|$ is at most $2^{O(nm)}/((K)!)^m$ (This will be formally proven in Section 1.3 Fact 19 of the Supplementary Information). As we choose $K = \mathsf{poly}(n)$, this is at most $2\ ^{\mathsf{poly}(n)}$ for every desired polynomial. □

1.2 Interpreting and extending Theorem 1

We have now established a worst-to-average case reduction with respect to the distribution $\mathcal{D}'_C \overset{\mathsf{def}}{=} C \times \mathcal{H}_{\mathcal{A},\theta,K}$. We now provide several sections to interpret this result. First, although we have changed the distribution over which average-case hardness is established, we show that hardness over the new distribution, Theorem 1, is *necessary* for Conjecture 14 to be true in Section 1.3 of the Supplementary Information. Second, we show the approximate version of this theorem (over the family of distributions $\mathcal{D}'_C$) is *equivalent* to the original conjecture (over $\mathcal{H}_\mathcal{A}$) – see Section 1.4 of the Supplementary Information. So an approximation-robust version of Theorem 1 would prove the original Conjecture 14 – which is precisely what we desire from our result. Hence Theorem 1 is precisely an analog of Theorem 13 for BosonSampling. Additionally, although our result is phrased in terms of exactly computing $\mathsf{p_0}(C')$, we show we can make this result robust to some fixed inverse exponential approximation error (see Section 1.7 of the Supplementary Information) – this follows from the same arguments used in BosonSampling [8]. So, to prove Conjecture 14, one only needs to improve the robustness.

At a high level, to see why our hardness result is necessary for Conjecture 14 to be true, consider the values of the parameters in Conjecture 14, which we state here:

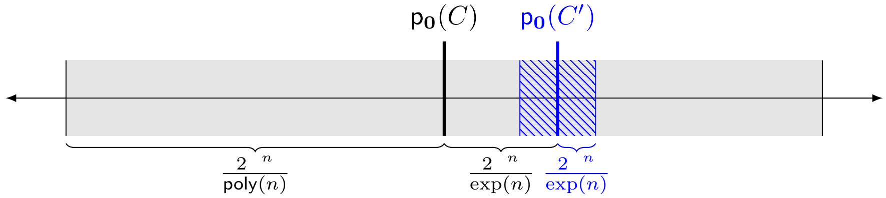
Conjecture 14 *There exists an architecture $\mathcal{A}$ so that the following task is #P-hard: Approximate $|\langle 0^n|\, C\, |0^n\rangle|^2$ to additive error $\pm\epsilon/2^n$ with probability $\frac{3}{4} + \frac{1}{\mathsf{poly}(n)}$ over the choice of $C \sim \mathcal{H}_\mathcal{A}$ in time* $\mathsf{poly}(n, 1/\epsilon)$.

Conjecture 14 assets that is it #P-hard to compute anything in an interval of radius $\frac{1}{2^n \mathsf{poly}(n)}$ around the point $\mathsf{p_0}(C)$ on average over the choice of C. Our result states that it is #P-hard to compute a different quantity $\mathsf{p_0}(C')$ on average, where C' is a truncated version of C (which we can also show remains true up to a tiny inverse-exponential amount of additive error). But this quantity $\mathsf{p_0}(C')$ is extremely close to $\mathsf{p_0}(C)$. The fact this hardness is *necessary* for Conjecture 14 essentially follows from the fact that this hardness interval is completely contained within the window of conjectured hardness. The fact that making it robust would be *sufficient* to prove Conjecture 14 essentially follows from that fact that $\mathsf{p_0}(C)$ and $\mathsf{p_0}(C')$ are so close to one another that a $2\ ^n/\mathsf{poly}(n)$ approximation to one is a $2\ ^n/\mathsf{poly}(n)$ approximation to the other and vice versa. This is illustrated in Supplementary Figure 3.

Furthermore, we note that this hardness result is generic. One can easily show our proof techniques apply to essentially any distribution D over quantum circuits over qubits with the following properties:

- The support of D contains instances which are worst-case #P-hard.
- D "scrambles" any worst-case instance back to D (or some distribution close in total variation distance).
- D is a continuous distribution over gates.

The necessity of the corresponding hardness result carries through as well. Therefore, our result is a generic tool for establishing hardness of distributions of randomly chosen circuits over qubits.


$\mathsf{p_0}(C)$
$\mathsf{p_0}(C')$
$\frac{2^{\ n}}{\mathsf{poly}(n)}$
$\frac{2^{\ n}}{\mathsf{exp}(n)}$
$\frac{2^{\ n}}{\mathsf{exp}(n)}$

Supplementary Figure 3: Our hardness result in the context of Conjecture 14. Our result establishes that outputting any number in the blue hashed interval centered around $\mathsf{p_0}(C')$ on average is #P-hard, while the original conjecture states that outputting any number in the larger grey interval around $\mathsf{p_0}(C)$ on average is #P-hard. Note: the two intervals of length $\frac{2^{\ n}}{\mathsf{exp}(n)}$ correspond to two different exponential functions.

1.3 Theorem 1 is necessary for Conjecture 14

To reflect on our result, Theorem 1 shows that a worst-to-average-case reduction is indeed possible with respect to a distribution over circuits that is close to the distribution $\mathcal{H}_{\mathcal{A}}$ we desire. Here we show that the hardness result established in Theorem 1 will be necessary to prove Conjecture 14. At a high level, this is because the discretized circuit amplitudes are exponentially close to those of the base circuit, and this perturbation falls within the approximation window of Conjecture 14.

Let us start by choosing some convenient notation. For the purposes of this section, let us fix an architecture $\mathcal{A}$ as well as parameters $\theta = \frac{1}{\mathsf{poly}(n)}$, and $K = \mathsf{poly}(n)$. Then, with respect to a fixed circuit C over this architecture, we denote the distribution $C \times \mathcal{H}_{\mathcal{A},\theta}$ as $\mathcal{D}_C$ (i.e., the corresponding θ perturbed Haar-distribution), and $C \times \mathcal{H}_{\mathcal{A},\theta,K}$ will be denoted $\mathcal{D}'_C$ (i.e., the corresponding (θ, K) truncated perturbed Haar-distribution). We also define the joint distribution of $\mathcal{D}_C$ and $\mathcal{D}'_C$, which we denote by $\mathcal{J}_C$. This is the distribution over pairs of circuits (C_1, C_2) generated by choosing independent Haar random gates $\{H_j\}_{j=1\ldots m}$ and using this choice to publish C_1 from $\mathcal{D}_C$ and C_2 from $\mathcal{D}'_C$, using the same choice of $\{H_j\}$. Then, the marginal of $\mathcal{J}_C$ on C_1 is $\mathcal{D}_C$ and on C_2 is $\mathcal{D}'_C$ but they are correlated due to the same choice of $\{H_j\}$. For simplicity of notation, we will often suppress the argument C and simply write $\mathcal{D}, \mathcal{D}', \mathcal{J}$.

Now we will show how to use the existence of an algorithm for computing probabilities of most circuits with respect to the $\mathcal{D}'$ to estimate probabilities of most circuits drawn from $\mathcal{H}_{\mathcal{A}}$. We introduce one more helpful definition for these results, namely:

Definition 16 *We say an algorithm $\mathcal{O}$ $(\ ,\epsilon)$-computes a quantity $p(x)$ with respect to a distribution F over inputs if:*

$$\Pr_{x\sim F}\left[p(x) \quad \epsilon \leq \mathcal{O}(x) \leq p(x) + \epsilon\right] \quad 1 \qquad .$$

In other words, the algorithm computes an estimate to the desired quantity with high-probability over instances drawn from F. In these terms, the main result of this section will be:

Theorem 17 *Suppose there exists an efficient algorithm $\mathcal{O}$ that for architecture $\mathcal{A}$, $(\epsilon,\ )$-computes the $\mathsf{p_0}(C')$ probability with respect to circuits $C' \sim \mathcal{D}'$, then there exists an efficient algorithm $\mathcal{O}'$ that $(\epsilon',\ ')$-computes the $\mathsf{p_0}(C')$ probability with respect to circuits $C' \sim \mathcal{H}_{\mathcal{A}}$, with $\epsilon' = \epsilon + 1/\mathsf{exp}(n)$ and $\ ' = \ + 1/\mathsf{poly}(n)$.*

Note that, as in the statement of Theorem 1, the algorithm $\mathcal{O}$ is assumed to work for $\mathcal{D}'$ for any choice of C (since $\mathcal{D}'$ is defined relative to a circuit C). From this, one has the following immediate corollary:

Corollary 18 *Conjecture 14 implies Theorem 1.*

Proof: If there is an algorithm exactly computing probabilities on average (i.e. with probability $\frac{3}{4} + \frac{1}{\mathsf{poly}(n)}$) over $\mathcal{D}'$, then there is an algorithm approximately computing probabilities on average over $\mathcal{H}_{\mathcal{A}}$. Therefore, if approximately computing probabilities on average over $\mathcal{H}_{\mathcal{A}}$ is #P-hard, then exactly computing probabilities on average over $\mathcal{D}'$ is #P-hard as well. Hence Conjecture 14 implies there exists some architecture $\mathcal{A}$ for which exactly computing probabilities on average over $\mathcal{D}'$ is #P-hard. In fact, Theorem 1 says something even stronger than this, as it applies to all architectures with worst-case #P-hardness. Therefore a weaker form of Theorem 1 is necessary for Conjecture 14 to be true. □

In other words, our main result is necessary for the quantum supremacy conjecture (Conjecture 14) to be true.

We start proving Theorem 17 by establishing two facts which relate the distributions of circuits drawn from the joint distribution $\mathcal{J}$. A natural interpretation of Facts 19 and 20 is as statements about the proximity of output probabilities and input distributions, respectively. Fact 19 states that the output probabilities of circuits drawn from the joint distribution $\mathcal{J}$ are effectively the same. Fact 20 states the perturbed distribution is essentially $\mathcal{H}_{\mathcal{A}}$ – therefore, choosing the inputs from $\mathcal{H}_{\mathcal{A}}$ or the perturbed distribution is immaterial.

Fact 19 *Let $\mathcal{A}$ be an architecture over circuits and C a circuit in the architecture. Let (C_1, C_2) be circuits drawn from $\mathcal{J}$. Then the zero probabilities of C_1 and C_2 are close; namely,*

$$|\mathsf{p_0}(C_1) - \mathsf{p_0}(C_2)| \leq 2^{-\mathsf{poly}(n)}.$$

Proof: By expanding the exponential as a Taylor series, we can express each gate $C_{1,j}$ and $C_{2,j}$ of C_1 and C_2, respectively, as

$$C_{1,j} = C_j H_j \left(\sum_{k=0}^{\infty} \frac{(-ih_j\theta)^k}{k!} \right); \qquad C_{2,j} = C_j H_j \left(\sum_{k=0}^{K} \frac{(-ih_j\theta)^k}{k!} \right).$$

Therefore, $C_{1,j} - C_{2,j} = C_j H_j \left(\sum_{k=K+1}^{\infty} \frac{(-ih_j\theta)^k}{k!} \right)$. We can apply the standard bound on Taylor series to bound $|\langle y_j|C_{1,j} - C_{2,j}|y_{j-1}\rangle| \leq \frac{\kappa}{K!}$ for some constant κ. Applying this to a Feynman path integral,

$$|\langle 0|C_1|0\rangle - \langle 0|C_2|0\rangle| \leq \sum_{y_1,\ldots,y_m} \prod_{j=1}^{m} \langle y_j|C_{1,j}|y_{j-1}\rangle - \prod_{j=1}^{m} \Bigg(\langle y_j|C_{2,j}|y_{j-1}\rangle \leq 2^{n(m-1)} \cdot \mathcal{O}\left(\frac{m\kappa}{K!}\right) = \frac{2^{O(nm)}}{(K!)^m}.$$

This proves that the amplitudes are close. As the amplitudes have norm at most 1, then the probabilities are at least as close. The result follows by a sufficiently large choice of $K = \mathsf{poly}(n)$. □

Fact 20 *Let $\mathcal{A}$ be an architecture on circuits with m gates and $C \in \mathcal{A}$ a circuit from that architecture. Then the distribution $\mathcal{H}_{\mathcal{A}}$ and $\mathcal{D}$ are $O(1/\mathsf{poly}(n))$ close in total variation distance.*

Note that $\mathcal{D}$ depends on C, and that Fact 20 holds for *any* choice of C.

Proof:

To prove this, we will show that for any particular gate of the circuit, the distributions induced by $\mathcal{H}_{\mathcal{A}}$ and $\mathcal{D}$ are $O(\theta)$ close in total variation distance. Then the additivity of total variation distance for independent events implies that the distributions are $O(m\theta)$-close (i.e. if D and D' are ϵ-close in total variation distance, then n independent copies of D are $n\epsilon$-close to n independent copies of D'). The result then follows from a suitably small choice of $\theta = 1/\mathsf{poly}(n)$.

Now consider the distributions $\mathcal{H}_{\mathcal{A}}$ and $\mathcal{D}$ on a single two-qubit gate. Since the Haar measure $\mathcal{H}$ is left-invariant, the distance between these is equivalent to the distance between $C \times \mathcal{H}$ and $\mathcal{D} = C \times \mathcal{H}_\theta$, where $\mathcal{H}_\theta$ is the perturbed Haar distribution on a single gate. Since total variation distance is invariant under left multiplication by a unitary, this is equivalent to the distance between $\mathcal{H}$ and $\mathcal{H}_\theta$.

Intuitively, the reason these are $O(\theta)$ close is as follows: consider a random rotation in $SO(3)$, vs. a random rotation in $SO(3)$ which has been "pulled back" towards the identity. By construction, the axes of rotations will be uniformly random over the sphere in both distributions. The only difference between the distributions lies in their angles of rotation – the former's angle of rotation is uniform in $[0, 2\pi]$ while the latter's is uniform in $[0, 2\pi(1-\theta)]$. These distributions over angles are clearly θ-close in total variation distance. This immediate implies these distributions over matrices are θ-close in total variation distance as well since matrices are uniquely defined by the eigenbasis and eigenvalues.

We can extend this logic to the two-qubit case as well. By construction the distributions $\mathcal{H}$ and $\mathcal{H}_\theta$ will be diagonal in a uniformly random basis U (since "pulling back" a matrix A by $e^{\ \ i\theta \log A}$ preserves the eigenbasis). Hence the only difference between these distributions lies in their distribution over eigenvalues. We will show their distribution over eigenvalues are $O(\theta)$ close in total variation distance, which will imply the claim. In particular, the distribution of eigenvalues $e^{i\theta_1}, e^{i\theta_2}, e^{i\theta_3}, e^{i\theta_4}$ of a two qubit gate drawn from $\mathcal{H}$ is given by the density function, due to Weyl (e.g. [40]),

$$\Pr\Big[\theta_i = \hat{\theta}_i\Big] \propto \prod_{i\neq j} \quad e^{i\hat{\theta}_i} \quad\quad e^{i\hat{\theta}_j} \quad {}^2 .$$

In contrast the distribution over eigenvalues of a two-qubit gate drawn from $\mathcal{H}_\theta$ is

$$\Pr\Big[\theta_i = \hat{\theta}_i\Big] \propto \begin{cases} 0 & \exists i : \hat{\theta}_i \quad 2\pi(1 \quad \theta) \\ \prod_{i\neq j} \quad e^{i\hat{\theta}_i} \quad\quad e^{i\hat{\theta}_j} \quad {}^2 & o.w. \end{cases}$$

One can easily compute that the total variation distance between these measures is $O(\theta)$, which implies the claim. This simply uses the fact that the above density function is smooth and Lipschitz, so a version of the same density function which has been "shifted" by θ is $O(\theta)$ close in total variation distance. □

Armed with these facts we are now ready to prove Theorem 17. We divide the proof into two steps, encapsulated into two lemmas (Lemmas 21, 22). In the first, we show how to use an algorithm that works on average over circuits drawn from $\mathcal{D}'$ to get an algorithm that works on average over pairs of circuits drawn from $\mathcal{H}$ and $\mathcal{D}$.

Lemma 21 *Suppose there exists an algorithm $\mathcal{O}$ that for any circuit C from a fixed architecture $\mathcal{A}$ takes as input a circuit C_2 sampled from $\mathcal{D}'$ and $(\epsilon, \ )$-computes the $\mathsf{p_0}(C_2)$ probability. Then there exists an algorithm $\mathcal{O}'$ that receives as input a random circuit $C \sim \mathcal{H}_\mathcal{A}$ as well as a sample $C_1 \sim \mathcal{D}$ and $(\epsilon', \ )$-computes the $\mathsf{p_0}(C_1)$ probability, where $\epsilon' = \epsilon + 1/\exp(n)$.*

Proof: This lemma is primarily a consequence of Fact 19. Our objective in the proof will be to develop an algorithm $\mathcal{O}'$ that, given a circuit C_1 from the $\mathcal{H}_{\mathcal{A},\theta}$ infers the corresponding circuit C_2 from $\mathcal{H}_{\mathcal{A},\theta,K}$. Once it does this, it simply returns the output of $\mathcal{O}$ run on input C_2.

More formally, consider an algorithm $\mathcal{O}'$ that is given as input C, as well as a pair of circuits $(C_1, C_2) \sim \mathcal{J}$, where $\mathcal{J}$ is the joint distribution with respect to C. Then $\mathcal{O}'$ runs $\mathcal{O}$ on input C_2. Clearly, from Fact 19, the output probabilities of C_1 and C_2 are exponentially close, so we can see that $\mathcal{O}'$ $(\epsilon + 1/\exp(n), \ )$-computes the quantity $\mathsf{p_0}(C_1)$.

Now by averaging over C, we see that in fact $\mathcal{O}'$ $(\epsilon + 1/\exp(n), \ )$-computes $\mathsf{p_0}(C_1)$ with respect to a distribution over triplets of circuits (C, C_1, C_2) in which $C \sim \mathcal{H}_\mathcal{A}$ and the pair (C_1, C_2) is distributed via the corresponding joint distribution $\mathcal{J}$. Next notice that instead of receiving the triplet of inputs (C, C_1, C_2), $\mathcal{O}'$ could simply have received a circuit $C \sim \mathcal{H}_\mathcal{A}$ and a circuit $C_1 \sim \mathcal{D}$. This is because it can infer the truncated circuit C_2 directly from C and C_1, by left-multiplying C_1 by $C^\dagger$ to obtain the element drawn from $\mathcal{H}_{\mathcal{A},\theta}$. As θ is fixed beforehand, the algorithm can then deduce the corresponding element drawn from $\mathcal{H}_\mathcal{A}$ with probability 1 by simply diagonalizing each gate and stretching the eigenvalues by $1/(1 \quad \theta)$. It can then compute the truncated Taylor series to obtain C_2. The Lemma follows. □

Next, we show how to use this new algorithm $\mathcal{O}'$ that works on average over pairs of circuits drawn from $\mathcal{H}_\mathcal{A}$ and $\mathcal{D}$ to get an algorithm $\mathcal{O}''$ that works on average over circuits drawn from $\mathcal{H}_\mathcal{A}$.

Lemma 22 *Suppose there exists an algorithm $\mathcal{O}'$ that takes as input a random circuit $C \sim \mathcal{H}_\mathcal{A}$ from a fixed architecture $\mathcal{A}$ as well as a circuit $C_1 \sim \mathcal{D}$, and $(\epsilon, \ )$-computes the $\mathsf{p_0}(C_1)$ probability. Then there exists an algorithm $\mathcal{O}''$ that $(\epsilon, \ ')$-computes the $\mathsf{p_0}(C)$ probability with respect to input circuits $C \sim \mathcal{H}_\mathcal{A}$, with $\ ' = \ \ + 1/\mathsf{poly}(n)$.*

Proof: This lemma is a direct consequence of Fact 20. In particular, Fact 20 implies that the input distribution to the algorithm $\mathcal{O}'$, in which the first input circuit is drawn from $\mathcal{H}_{\mathcal{A}}$ and the second is drawn from $\mathcal{D}$, is $1/\mathsf{poly}(n)$-close in total variation distance to the distribution over pairs of circuits independently drawn from $\mathcal{H}_{\mathcal{A}}$ which we refer to as $\mathcal{H}_{\mathcal{A}}^{(2)}$. This crucially relies on the property that Fact 20 is true for all C, i.e. $\mathcal{D} = \mathcal{D}_C$ (which depends on C) is close in total variation distance to $\mathcal{H}_{\mathcal{A}}$ for any choice of C.

To see this, let $A(C, C_1)$ be the probability density function (PDF) for the distribution on inputs to $\mathcal{O}'$, and let $B(C, C_1)$ be the PDF for $\mathcal{H}_{\mathcal{A}}^{(2)}$ (when integrated with respect to the Haar measure on pairs (C, C_1)). Then we have that

$$\begin{aligned} ||A \quad B|| &= \frac{1}{2}\iint dC dC_1 |A(C,C_1) \quad B(C,C_1)| = \frac{1}{2}\iint dC dC_1 |\mathcal{D}_C(C_1) \quad 1| \\ &= \iint dC \frac{1}{2} \iint dC_1 |\mathcal{D}_C(C_1) \quad 1| \\ &= \iint dC ||\mathcal{D}_C \quad \mathcal{H}_{\mathcal{A}}|| \leq O(1/\mathsf{poly}(n)) \end{aligned}$$

Where the first line follows from the fact that the PDF of the Haar measure is 1, and both measures' marginal distribution on the first input is Haar, the second line follows from splitting the joint integral into two integrals, and the third from Fact 20 (which holds for all C) as well as the convexity of integration.

Note total variation distance can be interpreted as the supremum over events of the difference in probabilities of those events. Considering the event that $\mathcal{O}'$ is approximately correct in its computation of $\mathsf{p_0}(C_1)$, this means if $\mathcal{O}'$ is run on inputs from the distribution $\mathcal{H}_{\mathcal{A}}^{(2)}$ instead of from $C \sim \mathcal{H}_{\mathcal{A}}$ and $C_1 \sim \mathcal{D}$, it will still be correct with high probability. So $\mathcal{O}'$ will $(\epsilon, \quad + 1/\mathsf{poly}(n))$-compute $\mathsf{p_0}(C_1)$ with respect to this new distribution $\mathcal{H}_{\mathcal{A}}^{(2)}$. Now these two input circuits are independently drawn, and so $\mathcal{O}'$ can discard the unused input circuit (i.e., the new algorithm $\mathcal{O}''$ merely needs to run this modified $\mathcal{O}'$ on its input circuit). We arrive at our Lemma. □

The results from Lemmas 21 and 22 together prove Theorem 17.

1.4 Approximate version of Theorem 1 is equivalent to Conjecture 14

We have shown that Theorem 1 is necessary for Conjecture 14. In this section we show a further nice property of Theorem 1. Namely, we show that the approximate version of Theorem 1 is equivalent to Conjecture 14, by proving the converse of Theorem 17. At a high level this follows because the truncated and non-truncated amplitudes are inverse exponentially close to one another, so an inverse polynomial approximation to one is an inverse-polynomial approximation to the other and vice versa. This implies that proving an approximation robust version of our Theorem 1 would in fact prove the original Conjecture 14 as desired. Hence our result can be seen one particular exact version of Conjecture 14, and is analogous to what is known in the case of Permanents and BosonSampling [8].

Theorem 23 *Suppose there exists an efficient algorithm $\mathcal{O}$ that $(\epsilon, \quad)$-computes the $\mathsf{p_0}(C)$ probability with respect to circuits C drawn from $\mathcal{H}_{\mathcal{A}}$. Then there exists an efficient algorithm $\mathcal{O}'$ that for architecture $\mathcal{A}$ and any fixed circuit C, given C as well as C' drawn from $\mathcal{D}'_C$, $(\epsilon', \quad ')$-computes the $\mathsf{p_0}(C')$ probability, where $\epsilon' = \epsilon + 1/\mathsf{exp}(n)$ and $\quad ' = \quad + 1/\mathsf{poly}(n)$.*

Corollary 24 *The following two statements are equivalent:*

- *It is #P-hard to $(\epsilon, \quad)$-compute the $\mathsf{p_0}(C)$ probability with respect to circuits C drawn from $\mathcal{H}_{\mathcal{A}}$, for any $\epsilon, \quad = \Omega(1/\mathsf{poly}(n))$. (i.e., Conjecture 14)*

- *It is #P-hard to $(\epsilon, \quad)$-compute the $\mathsf{p_0}(C')$ probability with respect to circuits C' drawn from any distribution in $\mathcal{D}'_C$, for any $\epsilon, \quad = \Omega(1/\mathsf{poly}(n))$.*

Proof: The proof closely resembles that of Theorem 17 - it makes careful use of Facts 19 and 20. Suppose $\mathcal{O}$ that $(\epsilon, \ )$-computes the $\mathsf{p_0}(C)$ probability with respect to circuits C drawn from $\mathcal{H}_{\mathcal{A}}$. Now note that by Fact 20, for any C, $\mathcal{D}$ is $O(1/\mathsf{poly}(n))$ close to $\mathcal{H}_{\mathcal{A}}$ in total variation distance. Again as total variation distance is a supremum over events of differences in probability, this means that if $\mathcal{O}$ is instead fed instances from $\mathcal{D}$, it will remain correct with high probability (by considering the event the computation is correct). So $\mathcal{O}$ will $(\epsilon, \ + O(1/\mathsf{poly}(n)))$-compute the $\mathsf{p_0}(C_1)$ probability with respect to circuits C_1 drawn from any distribution $\mathcal{D}_C$.

Now suppose we create an oracle $\mathcal{O}'$ which takes as input circuits C_1, C_2 drawn from the joint distribution $\mathcal{J}(C)$, and runs $\mathcal{O}$ on C_1. Now by construction this oracle $\mathcal{O}'$ will $(\epsilon, \ + 1/\mathsf{poly})$-compute the $\mathsf{p_0}(C_1)$. But by Fact 19 $\mathsf{p_0}(C_1)$ is very close to $\mathsf{p_0}(C_2)$, and therefore $\mathcal{O}'$ will $(\epsilon + 1/\mathsf{exp}(n), \ + O(m\theta))$-compute the $\mathsf{p_0}(C_2)$. But now note that one can infer the circuit C_1 from the value of C_2, as in the proof of Theorem 17 (again one would simply left multiply C_2 by $C^\dagger$, diagonalize, infer the values of the rotation angles $\ _i$ of the corresponding value of $\mathcal{H}_{\mathcal{A},\theta}$, and then output C_1.) Hence one can remove C_1 as input to the circuit, and obtain an oracle which approximately computes $\mathsf{p_0}(C_2)$ given arbitrary C and C_2 drawn from $\mathcal{D}'_C$ by inferring C_1 and running $\mathcal{O}$ on it. The theorem follows. □

1.5 Sampling implies average-case approximations in the polynomial hierarchy

In this section, we explain why Conjecture 14 implies quantum supremacy for RCS. In particular, we show that such an efficient classical algorithm for RCS would have surprising complexity consequences. This section will be very similar to analogous results in earlier work (see e.g., [8, 26, 27]).

That is, we show that the following algorithm which we call an approximate sampler, is unlikely to exist:

Definition 25 (Approximate sampler) *An approximate sampler is a classical probabilistic polynomial-time algorithm that takes as input a description of a quantum circuit C, as well as a parameter ϵ (specified in unary) and outputs a sample from a distribution D'_C such that*

$$||D_C \quad D'_C|| \le \epsilon$$

where D_C is the outcome distribution of the circuit C and the norm is total variation distance.

We note that this definition requires the approximate sampler to work for *all* circuits C. All of the arguments in this section would hold even if the approximate sampler worked on *most* circuits C drawn from $\mathcal{H}_{\mathcal{A}}$. While this latter definition better corresponds to the definition of RCS, we will analyze the former to simplify our presentation.

Our main result will connect the existence of an approximate sampler to an algorithm which will estimate the probabilities of most random circuits drawn from $\mathcal{H}_{\mathcal{A}}$, in the following sense:

Definition 26 (Average-case approximate solution) *An algorithm $\mathcal{O}$ is an average-case approximate solution to a quantity $p(x)$ with respect to an input distribution $\mathcal{D}$ if:*

$$\Pr_{x\sim\mathcal{D}}\left[\ \mathcal{O}(1^{1/\epsilon}, 1^{1/\ }, x) \quad p(x)\ \le \frac{\epsilon}{2^n}\right]\Bigg(\ 1 \quad .$$

In other words, an average-case approximate solution outputs a good estimate to the desired quantity for most random inputs but might fail to produce any such estimate for the remaining inputs.

More formally, the main theorem of this section, Theorem 28, proves that the existence of an approximate sampler implies the existence of an average-case approximate solution for computing the $\mathsf{p_0}(C)$ probability of a random circuit $C \sim \mathcal{H}_{\mathcal{A}}$. This average-case approximate solution will run in probabilistic polynomial time with access to an NP oracle. The main theoretical challenge in quantum supremacy is to give evidence that such an algorithm does not exist. This would certainly be the case if the problem was #P-hard, or as hard as counting the number of solutions to a boolean formula. Such a conjecture lies at the heart of all current supremacy proposals. More formally, this conjecture is:

Conjecture 14 *There exists a fixed architecture $\mathcal{A}$ so that computing an average-case approximate solution to $\mathsf{p_0}(C)$ with respect to $\mathcal{H}_{\mathcal{A}}$ is #P-hard.*

We now show how Conjecture 14 would rule out a classical approximate sampler for RCS, under well-believed assumptions. Specifically, assuming this conjecture is true, Theorem 28 tells us that an approximate sampler would give an algorithm for solving a #P-hard problem in $\mathsf{BPP}^{\mathsf{NP}}$. Now, $\mathsf{BPP}^{\mathsf{NP}}$ is known to be in the third-level of the PH (see e.g., [41]). In other words, $\mathsf{BPP}^{\mathsf{NP}} \subseteq \Sigma_3$. On the other hand, a famous theorem of Toda tells us that all problems solvable in the PH can be solved with the ability to solve #P-hard problems. That is, $\mathsf{PH} \subseteq \mathsf{P}^{\#\mathsf{P}}$ [42]. Putting everything together, we have that an approximate sampler would imply that $\mathsf{PH} \subseteq \Sigma_3$, a collapse of the PH to the third-level, a statement that is widely conjectured to be false (e.g., [43, 44]).

Finally, we prove Theorem 28. The proof utilizes a classic theorem by Stockmeyer [39], which we state here for convenience.

Theorem 27 (Stockmeyer [39]) *Given as input a function $f : \{0,1\}^n \to \{0,1\}^m$ and $y \in \{0,1\}^m$ there is a procedure that runs in randomized time $\mathsf{poly}(n, 1/\epsilon)$ with access to a NP^f oracle that outputs an α such that*

$$(1 \quad \epsilon)p \leq \alpha \leq (1+\epsilon)p \text{ for } p = \Pr_{x \sim \mathcal{U}(\{0,1\}^n)}[f(x) = y].$$

In the context of this work, the primary consequence of Stockmeyer's theorem is that we can use an NP oracle to get a multiplicative estimate to the probability of any outcome of an approximate sampler, by counting the fraction of random strings that map to this outcome. Using this idea we prove:

Theorem 28 *If there exists an approximate sampler $\mathcal{S}$ with respect to circuits from a fixed architecture $\mathcal{A}$, there also exists an average-case approximate solution in $\mathsf{BPP}^{\mathsf{NP}^{\mathcal{S}}}$ for computing the $\mathsf{p_0}(C)$ probability for a random circuit C drawn from $\mathcal{H}_{\mathcal{A}}$.*

Proof: We start by proving a related statement, which says that if we can sample approximately from the outcome distribution of any quantum circuit, we can approximate most of the output probabilities of all circuits C. This statement, unlike the Theorem 28, is architecture-agnostic.

Lemma 29 *If there exists an approximate sampler $\mathcal{S}$ then for any quantum circuit C, there exists an average-case approximate solution in $\mathsf{BPP}^{\mathsf{NP}^{\mathcal{S}}}$ for computing the $|\langle y| C |0\rangle|^2$ probability of a randomly chosen outcome $y \in \{0,1\}^n$.*

Proof: First fix parameters $\;, \epsilon > 0$. Then for any quantum circuit C, $\mathcal{S}(C, 1^{1/\eta})$ samples from a distribution η-close to the output distribution p of C. We denote this approximate outcome distribution by q. By Theorem 27, there exists an algorithm $\mathcal{O} \in \mathsf{BPP}^{\mathsf{NP}^{\mathcal{S}}}$ such that

$$(1 \quad)q_y \leq \mathcal{O}(C, y, 1^{1/\eta}, 1^{1/\;}) \leq (1 + \;)q_y.$$

Let $\tilde{q}_y = \mathcal{O}(C, y, 1^{1/\eta}, 1^{1/\;})$ for $\quad$ to be set later. Since q is a probability distribution, $\mathbb{E}(q_y) = 2^{\quad n}$. By Markov's inequality,

$$\Pr_y\left[q_y \quad \frac{k_1}{2^n} \quad\right] \leq \frac{1}{k_1}; \qquad \Pr_y\left[|q_y \quad \tilde{q}_y| \quad \frac{k_1}{2^n} \quad\right] \leq \frac{1}{k_1}.$$

Secondly, let $\Delta_y = |p_y \quad q_y|$. By assumption, $\sum_y \Delta_y = 2\eta$ so, therefore, $\mathbb{E}(\Delta_y) = 2\eta/2^n$. Another Markov's inequality gives

$$\Pr_y\left[\Delta_y \quad \frac{2k_2\eta}{2^n} \quad\right] \leq \frac{1}{k_2}.$$

With a union bound and a triangle inequality argument,

$$\Pr_y\left[|p_y \quad \tilde{q}_y| \quad \frac{k_1 + 2k_2\eta}{2^n} \quad\right] \leq \frac{1}{k_1} + \frac{1}{k_2}$$

Choose $k_1 = k_2 = 2/ \;$, $\; = (\epsilon\delta)/4, \eta = \; /2$. Then,

$$\Pr_y \left[|p_y \quad \tilde{q}_y| \quad \frac{\epsilon}{2^n} \right] \leq \; .$$

Therefore, for any circuit C, the algorithm $\mathcal{O}$ is an approximate average-case solution with respect to the uniform distribution over outcomes, as desired. □

Now we use the shared architecture constraint in the theorem statement to enable a so-called *hiding* argument. Hiding shows that if one can approximate the $|\langle y|C|0\rangle|^2$ probability for a random $y \in \{0,1\}^n$, the one can also approximate $\mathsf{p_0}(C)$ for a random C. This latter step will be crucial to our main result. In particular, both the anti-concentration property and our proof of average-case hardness of estimating circuit probabilities relies on considering a fixed output probability (see Sections 1.1 and 1.6 of the Supplementary Information).

To prove this, we rely on a specific property of $\mathcal{H}_{\mathcal{A}}$. This hiding property is that for any $C \sim \mathcal{H}_{\mathcal{A}}$, and uniformly random $y \in \{0,1\}^n$, $C_y \sim \mathcal{H}_{\mathcal{A}}$ where C_y is the circuit such that $\langle z| C_y |0\rangle = \langle z \quad y| C |0\rangle$. In other words, the distribution over circuits needs to closed under appending Pauli X gates to a random subset of output qubits.

Lemma 29 tells us that for any circuit C, an approximate sampler gives us the ability to estimate most output probabilities $\langle y| C |0\rangle$. If we instead restrict ourselves to $\mathcal{H}_{\mathcal{A}}$, we can think of this same algorithm $\mathcal{O}$ as giving an average-case approximate solution to $\mathsf{p_0}(C)$ with respect to the distribution generated by first choosing C from $\mathcal{H}_{\mathcal{A}}$ and then appending X gates to a uniformly chosen subset of the output qubits, specified by a string $y \in \{0,1\}^n$, since $\langle y| C |0\rangle = \langle 0| C_y |0\rangle$. Using the hiding property this is equivalent to an average-case approximate solution with respect to circuits C drawn from $\mathcal{H}_{\mathcal{A}}$, as stated in Theorem 28. □

1.6 Connecting with worst-case hardness and anti-concentration

Prior to this subsection, all of our results have been architecture agnostic– our worst-to-average case reduction in Section 1.1 of the Supplementary Information aims to reduce the presumed worst-case hardness of computing output probabilities of quantum circuits over a fixed architecture $\mathcal{A}$ to computing them on average over $\mathcal{H}_{\mathcal{A}}$.

Of course, for these results to be relevant to quantum supremacy, we need to establish that for the architectures $\mathcal{A}$ used in supremacy experiments, computing worst-case output probabilities is #P-hard. Then our worst-to-average-case reduction shows that computing average case probabilities for these experiments over $\mathcal{H}_{\mathcal{A}}$ is #P-hard – which is precisely what is necessary for the supremacy arguments of Section 1.1 of the Supplementary Information to hold. In this section, we will show that this requirement on $\mathcal{A}$ is quite mild. In particular, we will show that a candidate instantiation of RCS which is known to anti-concentrate – namely random quantum circuits on a 2D grid of depth $O(n)$ – easily satisfy this property. Therefore it is possible to have a single candidate RCS experiment which has both average-case #P-hardness as well as anti-concentration.

Such worst-case hardness can be established via the arguments of Bremner, Jozsa and Shepherd [10]. Although we will not summarize these standard arguments here, the key technical ingredient is demonstrating that quantum computations over this fixed architecture are universal. This will imply that the power of the corresponding complexity class supplemented with the ability to do post-selected measurements is equal in power to PostBQP = PP by a result of Aaronson [45]. That is, to show our worst-case hardness result it suffices to show that the class of problems solvable by circuits over a fixed architecture is equal to BQP. This can be established by standard results from measurement-based quantum computation involving universal resource states [46, 47, 48]. Roughly speaking, these results allow us to prepare a fixed state on a 2D grid and simulate any quantum circuit by performing a sequence of adaptive one-qubit measurements on this state. Combining these results immediately implies that if an architecture $\mathcal{A}$ is capable of generating one of these universal resource states, then $\mathcal{A}$ contains #P-hard instances – because one could simply post-select the measurement outcomes such that no adaptivity is required.

To be more formal, let us define some notation. Let $\mathcal{A} \subseteq \mathcal{A}'$ if the gates in $\mathcal{A}$ are a subset of those in $\mathcal{A}'$. Then if a circuit C is realizable in $\mathcal{A}$, then it is also realizable in $\mathcal{A}'$ - simply by setting those gates not in $\mathcal{A}$ to the identity (of course, one can also expand this definition to consider a one-qubit gate to be a subset of a two-qubit gate - as one can always set the two-qubit gate to be the identity tensor a one qubit gate.) Consider the "brickwork" state defined by Broadbent, Fitzsimons and Kashefi [49]. The brickwork state $|\ \ _{\text{brick}}\rangle$ is a universal resource state for measurement-based quantum computation, which has nice properties. In particular it can be prepared by a constant-depth quantum circuit C_{brick} on a 2D grid, where gates only act on nearest-neighbor qubits. Let $\mathcal{A}_{\text{brick}}$ be the architecture of C_{brick}, adding on space for one-qubit gates on every output qubit. Then $\mathcal{A}_{\text{brick}}$ is universal for quantum computation under post-selection by the above arguments. Therefore these prior results immediately yield the following Lemma:

Lemma 30 *For any architecture $\mathcal{A}$ such that $\mathcal{A}_{\text{brick}} \subseteq \mathcal{A}$, it is #$\mathsf{P}$-hard to exactly compute probabilities in $\mathcal{A}$.*

We first note that Lemma 30 is a *worst-case* hardness result: it states that exactly computing output probabilities for *all* circuits over $\mathcal{A}$ is #P-hard. However, the condition required to invoke Lemma 30 is extremely mild. It simply says that the architecture must contain a simple constant-depth nearest-neighbor circuit on a 2D grid as a subgraph. We now show that the mildness of this condition allow us to easily connect worst-case hardness to anti-concentration.

Let us first define anti-concentration and state why it is important in the context of quantum supremacy. Broadly speaking, anti-concentration is a statement about the distribution of probabilities. It states that *most* output probabilities are reasonably large.

Definition 31 (Anti-concentration) *For a fixed architecture $\mathcal{A}$, we say that RCS anti-concentrates on $\mathcal{A}$, if there exists constants $\kappa, \quad > 0$ so that:*

$$\Pr_{C\sim\mathcal{H}_\mathcal{A}}\Bigg[\Bigg(\mathsf{p_0}(C) \qquad \frac{1}{\kappa 2^n} \qquad 1 \qquad .$$

Crucially, this anti-concentration property allows us to reduce the hardness of average-case approximate solutions (which, by definition, approximate the desired circuit probability *additively*) to an average-case solution that approximates the solution *multiplicatively.* As such, we can at least ensure that these approximations are non-trivial, that is the signal is not lost to the noise. More formally,

Lemma 32 *For a fixed architecture $\mathcal{A}$ for which RCS anti-concentrates, if there exists an algorithm $\mathcal{O}$ that estimates $\mathsf{p_0}(C)$ to additive error $\pm\epsilon/2^n$ for a 1 fraction of $C \sim \mathcal{H}_\mathcal{A}$, then $\mathcal{O}'$ also can be used to estimate $\mathsf{p_0}(C)$ to multiplicative error $\epsilon \cdot \kappa$ for a 1 fraction of $C \sim \mathcal{H}_\mathcal{A}$.*

Proof: A rephrasing of the additive error assumption is $\Pr_{C\in\mathcal{H}_\mathcal{A}}\left[|\mathcal{O}(C) \quad \mathsf{p_0}(C)| > \frac{\epsilon}{2^n}\right] \leq \ $. We apply a union bound to argue that

$$\begin{aligned}\Pr_{C\in\mathcal{H}_\mathcal{A}}\left[|\mathcal{O}(C) \quad \mathsf{p_0}(C)| > \epsilon\kappa\mathsf{p_0}(C)\right] &\leq \Pr_{C\in\mathcal{H}_\mathcal{A}}\left[|\mathcal{O}(C) \quad \mathsf{p_0}(C)| > \frac{\epsilon}{2^n}\right] + \Pr_{C\in\mathcal{H}_\mathcal{A}}\left[\frac{\epsilon}{2^n} > \epsilon\kappa\mathsf{p_0}(C)\right] \\ &\leq \quad + \quad .\end{aligned}$$

□

Anti-concentration is known for random quantum circuits of depth $O(n)$. It is possible to show that this instantiation of RCS obeys the conditions of Lemma 30, and hence can exhibit both average-case hardness and anti-concentration simultaneously. More specifically, suppose that at each step one picks a random pair of nearest-neighbor qubits on a line, and applies a Haar random gate between those qubits, until the total depth of the circuit is $O(n)$. Prior work has established that such circuits are approximate quantum two-designs, i.e. they approximate the first two moments of the Haar measure on all n qubits of the system [18, 50]. This, combined with the fact that unitary two-designs are known to anti-concentrate (which was noted independently in multiple works [19, 31, 32]), implies that random circuits of depth $O(n)$ anti-concentrate.

These results immediately generalize to random circuits of depth $O(n)$ on a 2D grid. Note one can easily show that with probability $1 - o(1/\mathsf{poly}(n))$ over the choice of a random circuit in this model, the architecture of the circuit obeys Lemma 30. Hence, computing average-case probabilities over this random circuit model is #P-hard. Although here we are discussing average-case hardness over a random choice of architecture, this result easily follows from our reduction for a single architecture, since w.h.p. the architecture drawn is hard on average.

Therefore, random circuits of depth $O(n)$ on a 2D grid obtain both average-case hardness and anti-concentration. We note that it is conjectured that random circuits of depth $O(n^{1/2})$ on a 2D grid anti-concentrate as well [9]. If this conjecture is true then such circuits would also exhibit both anti-concentration and average-case hardness, as we only require constant depth to satisfy Lemma 30.

1.7 Approximate sampling to a fixed inverse exponential suffices

The statement of Theorem 1 describes the hardness of *exactly* computing the probabilities $\mathsf{p}_0(C')$ over the choice of C' from the distributions $\mathcal{D}'_C$. In this Section, we show that the statement can be improved to show that it remains #P-hard to *approximately* compute these probabilities to inverse exponential error 2^{-n^d}, for some fixed but arbitrary d (this parameter is set when one picks the degree of truncation K of the truncated circuit inputs.) This follows from the arguments outlined in Aaronson and Arkhipov, who noted the same fact for their context [8]. Combining this with Theorem 23, one sees that in order to prove Conjecture 14, one simply needs to improve this approximation tolerance from 2^{-n^d} to $1/\mathsf{poly}(n)$.

In particular, we can modify the proof of Theorem 1 as follows. Assume instead that our machine $\mathcal{O}$ approximately computes $\mathsf{p}_0(C')$ with tolerance 2^{-n^d} with probability $1 - 1/2mK^2$ when C' is drawn from $\mathcal{D}'_C$. Now, we instead randomly choose uniformly spaced $2mK+1$ values $\{\theta_\ell\}$ and interpolate the polynomial $\tilde{q}(\theta)$ from the points $\{(\theta_\ell, \mathcal{O}(\theta_\ell))\}$. Then with high probability, $\tilde{q}(\theta_\ell)$ will differ from $q(\theta_\ell)$ by at most 2^{-n^d}.

We now apply the technique of Aaronson and Arkhipov [8]. By an argument of Rakhmanov [51], we can further argue that for a range $[a, b]$ with $0 \leq a < b \leq 1/\mathsf{poly}(n)$ and $b - a = \Omega(1/\mathsf{poly}(n))$), that for every $\theta \in [a, b]$, $|q(\theta) - \tilde{q}(\theta)| \leq 2^{-n^{d'}}$ where d' is a constant dependent on d. In other words the difference between these polynomials is uniformly bounded within some (slightly narrower and taller) box. It is then a consequence of a lemma of Paturi [52] that $|q(1) - \tilde{q}(1)| \leq 2^{-n^c}$ – i.e., our polynomial interpolated on the noisy data is still a good approximation of our quantity of interest.

In [8], this argument is omitted but it is stated that it follows from a result of Paturi [52]. Although the argument of Paturi is insufficient by itself, we thank Aaronson for pointing out that it can be fixed by coupling it with Rakhmanov's result [51].

The rest of the proof follows that of Theorem 1. We note that we are unable to use the Berlekamp-Welch Algorithm [36] directly in this error-robust theorem as it is not robust to noise. This requires us to assume that the machine $\mathcal{O}$ produces an approximately correct estimate with probability $1 - 1/\mathsf{poly}(n)$. The same issue affects Aaronson and Arkhipov's result [8].

2 Verification of Random Circuit Sampling

2.1 Proof of Theorem 9

Theorem 9 *For every unitary U, there exists a distribution D_U such that, with probability $1 - o(1)$ over the choice of U from the Haar measure, $|D_U - p_U| \geq 0.99$, and yet $\mathsf{CE}(D_U, p_U)$ is $O(1/N^{\Theta(1)})$-close to ideal.*

Proof: (*Sketch*)

The basic idea is to consider a "rescaled" distribution on $1/k$ of the outputs for some sufficiently large integer k. That is, we will assign probability 0 to $1 - \frac{1}{k}$ fraction of the outputs, and multiply the probabilities on the remaining outputs by k. By construction, this has total variation distance roughly $1 - \frac{1}{k}$ from the ideal distribution and relatively small entropy. However, one can show it is essentially indistinguishable from the point of cross-entropy difference – that is the cross-entropy difference is exponentially close to the ideal.

To be more precise, consider listing the strings $x \in \{0,1\}^n$ as $x_1, \ldots, x_N$ in order of increasing $p_U(x)$. Label the strings x_i, $i = 1 \ldots N$, such that $i < j$ implies $p_U(x_i) < p_U(x_j)$. For simplicity, we will focus only on the "middle 99.9 percent" of the distribution, i.e., we will pick constants c_1, c_2 such that with high probability over the choice of U, 99.9 percent of probability mass is on x_i satisfying $\frac{c_1}{N} < p_U(x_i) \leq \frac{c_2}{N}$. We will consider values of i between i_{min}, the smallest i such that $\frac{c_1}{N} < p_U(x_i)$, and i_{max}, the largest i such that $p_U(x_i) < \frac{c_2}{N}$.

Now consider the distribution D_U defined as follows:

$$D_U(x_i) = \begin{cases} p_U(x_i) & i < i_{min} \quad \text{or} \quad i > i_{max} \\ p_U(x_i) + p_U(x_{i+1}) + \ldots + p_U(x_{i+k \quad 1}) & i_{min} \leq i \leq i_{max} \quad \text{and} \quad i = k\mathbb{N} \\ 0 & i_{min} \leq i \leq i_{max} \quad \text{and} \quad i \neq k\mathbb{N}. \end{cases}$$

It is not hard to see that the total variation distance between this distribution and the ideal distribution is $0.99(1 \quad \frac{1}{k})$ in expectation over the choice of U, and hence if $k = 500$ with high probability is more than 0.99 by standard concentration inequalities. Furthermore, a careful but straightforward calculation shows that the CE of this rescaled distribution D_U and p_U is exponentially close to the ideal score.

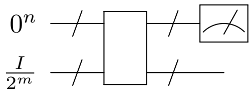
□

2.2 The importance of the "shape" of RCS output distributions

A basic property of RCS is that typical outcome distributions have a "Porter-Thomas", or exponential shape. That is, when one draws random unitary U, for any choice of constants $c_1 < c_2$ the number of x with $p_U(x)$ in the range $[c_1/N, c_2/N]$ will be roughly $N \int_{c_1}^{c_2} e^{\quad q} dq$ in expectation. Therefore, by concentration of measure, with high probability over the choice of U, the distribution induced by choosing a random x and sampling $p_U(x)$ is close to (a discretized version of) the Porter-Thomas, or exponential distribution.

By itself, such a Porter-Thomas distribution is not a signature of quantumness – below we give an example of a classical *physical* process resembling the physics of a noisy/decoherent quantum system, which can reproduce the "Porter-Thomas shape". The significance of the Porter-Thomas distribution lies in the fact that it has constant variation distance away from the uniform distribution. This fact facilitates statistical measures that take into account not only the shape of the distribution, but also the *identities* of which output strings correspond to which probabilities $p_U(x)$ under the ideal distribution.

Example: Consider a system of $n + m$ classical bits, the first n of which we will call the "system", and the second m of which we will call the "environment". Suppose that the system bits are initialized to 0, while the environment bits are chosen uniformly at random. Now suppose that one applies a uniformly random classical permutation to these $n + m$ bit strings (i.e., a random element   of $S_{2^{n+m}}$) and observes the first n system bits many times (while ignoring the environment bits) with the same choice of   but different settings of the environment bits. A diagram of this process is provided below in quantum circuit notation, but note this is a purely classical process.



Now we claim that the "shape" of this probability distribution closely resembles Porter-Thomas. Over the choice of   , each input string on $n + m$ bits is mapped to a uniformly random output string on $n + m$ bits (of which we only observe the first n bits). Therefore, this process resembles throwing 2^m balls (one for each possible setting of the environment bits) into 2^n bins (one for each possible output string of the system bits). We note that this approximation is valid only if m is sufficiently large (say $m = n$) – otherwise one would "notice" that   is a permutation rather than a random function, and the ball throws would not be approximately independent. This is analogous to the fact that the amplitudes of a random quantum state are only approximately independent because they are subject a normalization constraint. For simplicity, suppose we set $m = n$ (though we do not claim this choice is optimal). It is well known that in the large n limit, the distribution of the number of balls in each bin is close to the Poisson distribution with mean

$2^{m \quad n} = 1$ [53], i.e., the number of balls thrown into each bin is described as $\Pr[c = k] = \frac{1}{k!e}$ where c is the count in a particular bin. So normalizing by the number of balls, we see that for any output string x,

$$\Pr \left[p_{\text{Poisson}}(x) = \frac{k}{N} \right. = \frac{1}{k!e}.$$

We claim that this distribution is a natural classical imposter of Porter-Thomas. Since $k! = 2^{\Theta(k \log k)}$, this distribution is also (approximately) exponential. So this can be seen as a discretized version of Porter-Thomas, where the discretization resolution can be made finer by choosing larger m. Just as the Porter-Thomas distribution approximately describes the distribution on output probabilities of a quantum system under a random choice of U, here the Poisson distribution approximately describes the distribution on output probabilities of this classical system under a random choice of . And as the Porter-Thomas distribution is reproduced with unitary k-designs for sufficiently large k, here the Poisson statistics are reproduced when is chosen from a k-wise independent family for sufficiently large k. This follows because the number of bins with k balls is a kth order moment of the distribution.

2.3 The relationship between cross-entropy and HOG

The last section highlighted that any supremacy proposal based on outcome statistics must directly incorporate the *relationship* between outcome strings and their probabilities. One verification measure which takes this into account directly is Aaronson and Chen's Heavy Output Generation (or HOG). The task required of the quantum computer is simple: given a circuit description of a unitary U, output a list of strings such that a substantial fraction of them are "heavy" in the ideal output distribution:

Definition 33 ([28]) *Given as input a random quantum circuit U drawn from $\mathcal{H}_{\mathcal{A}}$, generate output strings $x_1, \ldots, x_k$, at least a 2/3 fraction of which have greater than the median probability in p_U.*

At first glace, this statistical test seems unrelated to cross-entropy. However, these measures are more similar than they first appear. Indeed, note than one can easily restate the HOG task as taking a certain expectation value over the device's output distribution.

Definition 34 *A family of distributions $\{D_U\}$ satisfies* Heavy Output Generation (HOG) *iff the following holds: Let*

$$\mathsf{HOG}(D_U, p_U) = \sum_{x \in \{0,1\}^n} D_U(x) \quad (p_U(x))$$

where $\ (z) = 1$ *if* $z \quad \frac{\ln 2}{N}$ *and* 0 *otherwise. Then the family is said to satisfy HOG if*

$$\mathbb{E}_{U \sim \mathcal{H}_{\mathcal{A}}} \mathsf{HOG}(D_U, p_U) \quad 2/3.$$

The quantity $\ln(2)/N$ is chosen because it is the median of Porter-Thomas. This is empirically measured as follows: pick a random U, obtain k samples $x_1, \ldots, x_k$ from the experimental device and compute:

$$H = \frac{1}{k} \sum_{i=1,\ldots,k} \quad (p_U(x_i)). \tag{5}$$

Phrased in this language, the similarities between cross-entropy and HOG (equations (1) and (5)) are readily apparent. Both are approximating the expectation value of some function of the ideal output probabilities $f(p_U(x_i))$ over the experimental output distribution. In the case of cross-entropy, $f(x) = \log(1/x)$. And in the case of HOG, $f(x) = \ (x)$. Both measures require only a small number of samples from the experimental device to compute to high accuracy by concentration of measure. Furthermore, cross-entropy and HOG directly verify more than just shape – rather they identify a particular relationship between outcome strings and their probabilities, and ensure that the quantum device tends to output "heavy" elements of the ideal distribution (with respect to some measure of heaviness).

While we have shown there is a natural noise model assumption under which classically scoring well on cross-entropy leads to a collapse of the polynomial hierarchy, no such connection is known for HOG. On the other hand, Aaronson and Chen connected the hardness of performing HOG to a complexity theoretic assumption known as QUATH. This is a nonstandard complexity conjecture and it remains open to connect the hardness of HOG to more standard complexity conjectures such as the non-collapse of the Polynomial Hierarchy. In short, these two measures verify quantum supremacy under two very different types of conjectures.

References

[1] Bernstein, E. & Vazirani, U. V. Quantum complexity theory. In Kosaraju, S. R., Johnson, D. S. & Aggarwal, A. (eds.) *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing, May 16-18, 1993, San Diego, CA, USA*, 11–20 (ACM, 1993). URL `http://doi.acm.org/10.1145/167088.167097`.

[2] Simon, D. R. On the power of quantum cryptography. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, 116–123 (IEEE Computer Society, 1994). URL `https://doi.org/10.1109/SFCS.1994.365701`.

[3] Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review* **41**, 303–332 (1999).

[4] Mohseni, M. *et al.* Commercialize quantum technologies in five years. *Nature* **543**, 171–174 (2017).

[5] Kandala, A. *et al.* Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **549**, 242 – 246 (2017). URL `http://dx.doi.org/10.1038/nature23879`.

[6] Zhang, J. *et al.* Observation of a many-body dynamical phase transition with a 53-qubit quantum simulator. *Nature* **551**, 601–604 (2017).

[7] Preskill, J. Quantum Computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018). URL `https://doi.org/10.22331/q-2018-08-06-79`.

[8] Aaronson, S. & Arkhipov, A. The computational complexity of linear optics. In *Proceedings of the forty-third annual ACM Symposium on Theory of Computing*, 333–342 (ACM, 2011).

[9] Boixo, S. *et al.* Characterizing quantum supremacy in near-term devices. *Nature Physics* **14**, 595–600 (2018). URL `https://doi.org/10.1038/s41567-018-0124-x`.

[10] Bremner, M. J., Jozsa, R. & Shepherd, D. J. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 459–472 (The Royal Society, 2010).

[11] Spring, J. B. *et al.* Boson sampling on a photonic chip. *Science* 798–801 (2012).

[12] Broome, M. A. *et al.* Photonic boson sampling in a tunable circuit. *Science* **339**, 794–798 (2013).

[13] Tillmann, M. *et al.* Experimental boson sampling. *Nature Photonics* **7**, 540–544 (2013).

[14] Crespi, A. *et al.* Integrated multimode interferometers with arbitrary designs for photonic boson sampling. *Nature Photonics* **7**, 545–549 (2013).

[15] Neville, A. *et al.* No imminent quantum supremacy by boson sampling. *arXiv:1705.00686* (2017).

[16] Clifford, P. & Clifford, R. The classical complexity of boson sampling. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, 146–155 (SIAM, 2018).

[17] Martinis, J. The quantum space race (2018). Plenary talk at Quantum Information Processing (QIP) 2018, Available at https://collegerama.tudelft.nl/Mediasite/Showcase/qip2018/Channel/qip-day3.

[18] Brandão, F. G. & Horodecki, M. Exponential quantum speed-ups are generic. *Quantum Information & Computation* **13**, 901–924 (2013).

[19] Hangleiter, D., Bermejo-Vega, J., Schwarz, M. & Eisert, J. Anticoncentration theorems for schemes showing a quantum speedup. *Quantum* **2**, 65 (2018).

[20] Terhal, B. M. & DiVincenzo, D. P. Adaptive quantum computation, constant depth quantum circuits and Arthur-Merlin games. *Quantum Information & Computation* **4**, 134–145 (2004).

[21] Morimae, T., Fujii, K. & Fitzsimons, J. F. Hardness of classically simulating the one-clean-qubit model. *Physical Review Letters* **112**, 130502 (2014).

[22] Farhi, E. & Harrow, A. W. Quantum supremacy through the quantum approximate optimization algorithm. *arXiv:1602.07674* (2016).

[23] Bouland, A., Mancinska, L. & Zhang, X. Complexity Classification of Two-Qubit Commuting Hamiltonians. In Raz, R. (ed.) *31st Conference on Computational Complexity (CCC 2016)*, vol. 50 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 28:1–28:33 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2016). URL `http://drops.dagstuhl.de/opus/volltexte/2016/5846`.

[24] Lipton, R. J. New directions in testing. *Distributed Computing and Cryptography* 191–202 (1991).

[25] Pastawski, F., Yoshida, B., Harlow, D. & Preskill, J. Holographic quantum error-correcting codes: Toy models for the bulk/boundary correspondence. *Journal of High Energy Physics* **2015**, 149 (2015).

[26] Fefferman, B. & Umans, C. On the power of quantum Fourier sampling. In *11th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2016, September 27-29, 2016, Berlin, Germany*, 1:1–1:19 (2016). URL `http://dx.doi.org/10.4230/LIPIcs.TQC.2016.1`.

[27] Bremner, M. J., Montanaro, A. & Shepherd, D. J. Average-case complexity versus approximate simulation of commuting quantum computations. *Physical Review Letters* **117**, 080501 (2016).

[28] Aaronson, S. & Chen, L. Complexity-theoretic foundations of quantum supremacy experiments. In O'Donnell, R. (ed.) *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, vol. 79 of *LIPIcs*, 22:1–22:67 (Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017). URL `https://doi.org/10.4230/LIPIcs.CCC.2017.22`.

[29] Bremner, M. J., Montanaro, A. & Shepherd, D. J. Achieving quantum supremacy with sparse and noisy commuting quantum computations. *Quantum* **1**, 8 (2017). URL `https://doi.org/10.22331/q-2017-04-25-8`.

[30] Morimae, T. Hardness of classically sampling the one-clean-qubit model with constant total variation distance error. *Physical Review A* **96**, 040302 (2017).

[31] Bouland, A., Fitzsimons, J. F. & Koh, D. E. Complexity Classification of Conjugated Clifford Circuits. In Servedio, R. A. (ed.) *33rd Computational Complexity Conference (CCC 2018)*, vol. 102 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 21:1–21:25 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2018). URL `http://drops.dagstuhl.de/opus/volltexte/2018/8867`.

[32] Mann, R. L. & Bremner, M. J. On the complexity of random quantum computations and the Jones polynomial. *arXiv:1711.00686* (2017).

[33] Harrow, A. W. & Low, R. A. Random quantum circuits are approximate 2-designs. *Communications in Mathematical Physics* **291**, 257–302 (2009). URL `https://doi.org/10.1007/s00220-009-0873-6`.

[34] Neill, C. *et al.* A blueprint for demonstrating quantum supremacy with superconducting qubits. *Science* **360**, 195–199 (2018).

[35] Boixo, S., Smelyanskiy, V. N. & Neven, H. Fourier analysis of sampling from noisy chaotic quantum circuits. *arXiv:1708.01875* (2017).

[36] Welch, L. & Berlekamp, E. Error correction for algebraic block codes (1986). URL `https://www.google.com/patents/US4633470`. US Patent 4,633,470.

[37] Gemmell, P., Lipton, R., Rubinfeld, R., Sudan, M. & Wigderson, A. Self-testing/correcting for polynomials and for approximate functions. In *Proceedings of the Twenty-third Annual ACM Symposium on Theory of Computing*, STOC '91, 33–42 (ACM, New York, NY, USA, 1991). URL `http://doi.acm.org/10.1145/103418.103429`.

[38] Valiant, L. The complexity of computing the permanent. *Theoretical Computer Science* **8**, 189 – 201 (1979). URL `http://www.sciencedirect.com/science/article/pii/0304397579900446`.

[39] Stockmeyer, L. On approximation algorithms for #P. *SIAM Journal on Computing* **14**, 849–861 (1985). URL `https://doi.org/10.1137/0214060`. `https://doi.org/10.1137/0214060`.

[40] Diaconis, P. & Shahshahani, M. On the eigenvalues of random matrices. *Journal of Applied Probability* 49–62 (1994).

[41] Lautemann, C. BPP and the Polynomial Hierarchy. *Information Processing Letters* **17**, 215–217 (1983).

[42] Toda, S. PP is as hard as the polynomial-time hierarchy. *SIAM J. Comput.* **20**, 865–877 (1991). URL `http://dx.doi.org/10.1137/0220053`.

[43] Karp, R. M. & Lipton, R. J. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing*, STOC '80, 302–309 (ACM, New York, NY, USA, 1980). URL `http://doi.acm.org/10.1145/800141.804678`.

[44] Boppana, R. B., Hastad, J. & Zachos, S. Does co-NP have short interactive proofs? *Inf. Process. Lett.* **25**, 127–132 (1987). URL `http://dx.doi.org/10.1016/0020-0190(87)90232-8`.

[45] Aaronson, S. Quantum computing, postselection, and probabilistic polynomial-time. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* **461**, 3473–3482 (2005). URL `http://rspa.royalsocietypublishing.org/content/461/2063/3473`. `http://rspa.royalsocietypublishing.org/content/461/2063/3473.full.pdf`.

[46] Raussendorf, R. & Briegel, H. J. A one-way quantum computer. *Physical Review Letters* **86**, 5188 (2001).

[47] Raussendorf, R., Browne, D. E. & Briegel, H. J. Measurement-based quantum computation on cluster states. *Physical Review A* **68**, 022312 (2003).

[48] Briegel, H. J., Browne, D. E., Dür, W., Raussendorf, R. & Van den Nest, M. Measurement-based quantum computation. *Nature Physics* **5**, 19–26 (2009).

[49] Broadbent, A., Fitzsimons, J. & Kashefi, E. Universal blind quantum computation. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, 517–526 (IEEE, 2009).

[50] Brandão, F. G., Harrow, A. W. & Horodecki, M. Local random quantum circuits are approximate polynomial-designs. *Communications in Mathematical Physics* **346**, 397–434 (2016).

[51] Rakhmanov, E. A. Bounds for polynomials with a unit discrete norm. *Annals of Mathematics* **165**, 55–88 (2007). URL `http://www.jstor.org/stable/20160024`.

[52] Paturi, R. On the degree of polynomials that approximate symmetric boolean functions (preliminary version). In *Proceedings of the Twenty-fourth Annual ACM Symposium on Theory of Computing*, STOC '92, 468–474 (ACM, New York, NY, USA, 1992). URL `http://doi.acm.org/10.1145/129712.129758`.

[53] Motwani, R. & Raghavan, P. *Randomized algorithms* (Chapman & Hall/CRC, 2010).