Random Circuits Sampling on IBM Qiskit devices

Yue Shi[†], Hao Li[‡]

December 2019

Contents

[†]Department of Physics, NYU, New York, NY, USA. *email: fred.shi@nyu.edu

[‡]Department of Physics, NYU, New York, NY, USA. *email: hl3270@nyu.edu

1 Introduction

1.1 Quantum Supremacy

The exponential speedup advantages quantum computers showed have given rise to a potential almighty computing model that can solve problems well beyond classical computers' reach. Mathematicians have come up with a impressive amount of quantum algorithms that could solve problems which can take the fastest classical computers infinitely many years to solve, such as factoring large numbers, finding generators for a large abelian group, and simulating complicated quantum systems [**1**, **2**, **3**].

However, limited by the quantum hardware we have, even the state of art quantum computers, with the exponential speedup quantum algorithms, are too noisy and unstable to solve problems that classical computers cannot solve. More importantly, it seems that a large and stable quantum computer is far from our current capabilities and may not be realized in the near future. Thus, an interesting question is raised: with the quantum computers we had in this NISQ (Noise Intermediate-Scale Quantum device) era, is there any problem our quantum computers can do while classical computers cannot do?

This idea is purposed by John Preskill, referred as quantum supremacy. From our understanding, such problem should be noise tolerant. In other words, it should be looking for some approximations rather than the exact answers. Also, it should be exponentially hard for classical computers, such that classical computers cannot even approximately the answer with polynomial time.

1.2 Sampling Problems and Boson Sampling

Sampling problems are those problems which output random numbers according to a particular probability distribution [**4**]. As for classical computers, we start with uniform distribution and then tries to transform the uniform distribution to the desired distribution. But for quantum computers, due to the nature of superposition, we can evolve the quantum state to the desired distribution and then measure to sample. As a result, we can evolve the quantum state to a probability distribution that classical computers do not know unless it simulates the evolution, which takes exponential time complexity.

This idea has been introduced by S. Aaronson and A. Arkhipov in 2011 [**5**], which they purposed sampling from identical bosons scattered by a linear interferometer, referred as boson sampling. More interestingly, strong evidence has shown that classical computer cannot even approximate the boson sampling problem in polynomial time, as it is related to the evaluation of permanents of matrices with complex entries, which is a well-known NP problem [**5**]. Thus if we can use a quantum computer, even with noise, to simulate the evolution of the bosons, we can get an approximation that classical computer cannot work out in polynomial time. This makes sampling problems an ideal candidate for demonstrating quantum supremacy. Experimental effort has been put in the photonic systems, in order to build a system large enough that no classical computers can calculate the probability distribution given the exponential time growth for simulating the system [**6**].

1.3 Random Circuits Sampling

Another sampling problem, referred as random circuits sampling, is also believed to be one of the candidates for showing quantum supremacy. It uses the delocalization of quantum states under evolution, a topic studied in the field of quantum chaos, to create a probability distribution that classical computer cannot work out in polynomial time. (We will elaborate the physics in the next section.) Similar to boson sampling, approximating the state of delocalized state is believed to be impossible in polynomial time for classical computers. But as this problem can be implemented on a universal quantum computer (gate based model), and is capable of showing quantum supremacy with 50 qubits with noise, which is roughly the state of the art universal quantum computer (both in the superconducting systems and ion trap systems) can have, it receives much attention and is well studied. Recently Google has claimed to reach "quantum supremacy"using their Sycamore 52 transmon superconducting device, where they show that their quantum computers can approximate this evolution with some level of correctness, while simulating (or approximating) the evolution with classical computers is beyond the maximal computing power we have [**7**].

1.4 Motivations

Being one of the most promising quantum computing hardware, superconducting qubit system has shown great capability and scalability for doing quantum computation. While Google was trying to reach "quantum supremacy"on their Sycamore 52 qubit device, IBM also claimed to develop their own 50 qubit system with similar coherence and accuracy. Therefore, it is interesting to see if we can repeat Google's

results on the slightly different IBM device, comparing and cross-checking the results. Unfortunately, we do not have the access to the 50 qubits devices. But IBM has shared us a couple of toy systems by the Qiskit toolbox [**8**], ranging from 1 qubit to 16 qubits to play with. Although 16 qubits system definitely cannot show "quantum supremacy", we believe it is still interesting to observe "quantum chaos"from the IBM superconducting devices and compare the results with the Google's paper.

Also, due to the user-friendly nature of the Qiskit toolbox, running programs on the state of the art quantum computers/superconducting system becomes convenient. As a result, we hope our attempts on the IBM devices with Qiskit toolbox can serve as some useful experience for the community if further investigations (on quantum chaos, quantum complexity, quantunm supremacy, etc.) are planned to be done using the IBM devices. In this report, we will first introduce the physics for quantum chaos in section 2, then explain what criteria Google used to show quantum chaos and what benchmarks they used to support their "quantum supremacy"results in section 3. We will run our codes for generating quantum chaos on the 5 and 15 qubits system on both the simulators and actual quantum computers in section 4. And lastly we will describe the experimental results and compare them with the ones from Google's paper in section 5.

2 Quantum Chaos and Delocalization

2.1 Quantum Chaos

In classical chaotic system, we know that the dynamic of a system is extremely sensitive to its initial condition. However, in quantum domain, the notion of phase-space trajectories, which classical chaotic system mainly studies on, lose its meaning due to the uncertainty principle. However, it is natural to search for some analogs in the quantum domain, and this field of research is referred as quantum chaos [**9**]. One of the topics quantum chaos studies is the so-called delocalization of quantum states: In some quantum many-body systems, a regime where the quantum states are exponentially sensitive to perturbation is observed. For such states, if we apply a small perturbation, the overlap of the perturbed and unperturbed states under evolution will decrease exponentially [**10**]. Such states are sometimes referred as the delocalized state, and are studied as an analog of the "quantum butterfly effect".

2.2 Computational Complexity of Delocalized State

Similar to classical chaotic systems, delocalized state is hard to track and approximate. Thus delocalized states are used for demonstrating quantum supremacy. Attempts have been made to show approximate the states is an NP-hard problem [**11**]. And the intuition follows as you cannot approximate a classical chaotic system unless you simulate the system with high accuracy, where in quantum domain is an exponentially hard operation.

2.3 Statistics of Delocalized States and Porter-Thomas Distribution

A very interesting feature of the delocalized states is its statistics. As the delocalized state is chaotic, it should be ergodic. In other words, the state will have visited the entire Hilbert space after a long time evolution, which again reflects the hardness of approximating the system. Also, its ergodic characteristics makes the system sufficiently complex, thus introducing some generic properties when looking at its statistics. Such generic properties in the non-commutative regime are first studied by Wigner, and later developed into the random matrix theory, which studies the "local universality"of different systems. In other words, the statistics of the delocalized states can be explained as some generic properties of the system makes it converges to the same distribution, where is universal for many other systems as long as they meet some criteria that makes them complicated enough and not dominated by some correlation. An analogy can be drawn from the central limit theorem, where sampling from identical independent distribution can converge to a Gaussian.

One of the interesting statistics of delocalized states is the distribution of the probability amplitudes of every delocalized state in a fixed basis. It is believed that it follows a Porter-Thomas distribution, which is a chi-square distribution of degree of freedom 2. In other words, if we pick a delocalized state $|u\rangle = x_1 |x_1\rangle + x_2 |x_2\rangle + ...x_n |x_n\rangle$ and measure many enough $|x_i\rangle$, the histogram of the probability of getting some state $|x_i\rangle$, i.e. $p(x_i) = |x_i|^2$, should follow the following distribution [**11**]:

$$f(p) = N\mathrm{e}^{-Np} \tag{1}$$

while the normal χ_2^2 distrubution $f_2(x) = \frac{1}{2}\mathrm{e}^{-\frac{x}{2}}$ is just a special case. In another prospective, given a fixed probability p, the probability of getting a randomly picked

state $|x_i\rangle$ which satiesfies $p(x_i) \approx p$ should follow this Porter-Thomas distribution. From Eq. 1 we know that the higher the probability given, the fewer the states with a similar probability of getting are there. Here N is the dimension of the system.

The insights we gain from the statistics is that we can easily check if a state is delocalized or not by checking its statistics. If the probability amplitudes follow the Porter-Thomas distribution, then we know that it is a delocalized state and thus it cannot be approximated by the classic computers with polynomial time algorithms. We then can use quantum computers to simulate such delocalized state and sample from state by measurement efficiently, thus creating a sampling problem that classic computers cannot even approximate in polynomial time.

2.4 Generating Delocalized States

Delocalized states can be generated by the random circuits, where you apply random unitary gates to generate randomness, and random two-qubit gates (such as CNOT) to entangle the system on a general quantum computer [**10**], and evolve some unstructured, many-body interacting Hamiltonians that is non-integrable [**9**]. While Google has shown they can generate "quantum chaos"by the random unitary gates and CZ gates and provided robust numerics indicating that the probability amplitudes of its delocalized states follow Porter-Thomas distribution, we will run random unitary gates and CNOT gates on simulator and actual quantum computer in section 4, to show that we can create quantum chaos on the IBM device as well. We will also compare the time taken to reach quantum chaos and other benchmarkings of computation between the two systems (will be introduced in section 3) in section 5 as they have different hardware geometry and different sets of gates applied.

3 Fidelity and Cross-Entropy Benchmarking

3.1 Cross-Entropy

To demonstrate quantum supremacy, two criterions are required, the first of which is the computational complexity. As mentioned above, universal random circuit sampling will induce chaotic quantum evolution, which is believed hard to simulated classically, and and so are delocalized states, which can be generated by random circuit sampling. Therefore, if evidence of delocalized states can be observed in RQCs, then we may possibly imply that the complexity criterion is fulfilled. We have known that the distribution of the output follows Porter-Thomas distribution, but simply the histogram of the experimental measurement it self is not enough signify delocalized states, since there exist systems that are easy to simulate classically and their outputs satiesfy Porter-Thomas distribution [**11**]. Instead, the relationship between the theoretical output by classical simulation and the measured results should be take into account. So generally the verification of quantum supremacy requires classical simulations of large systems and is therefore computationally expensive.

To represent the difference of the expected distribution from classical simulation and the measured output, cross-entropy is introduced. As we have learnt in class, the entropy of a probability distribution $p(x_i)$ is

$$S(p(x_i)) = -\sum_{x_i} p(x_i) \log(p(x_i)) \tag{2}$$

Here log is the natural logarithm. The cross-entropy of two probability distributions $p(x_i)$ and $q(x_i)$ defined quite similarly but is instead a combination

$$S(p(x_i), q(x_i)) = -\sum_{x_i} p(x_i) \log(q(x_i)) \tag{3}$$

If an RQCs experiment is performed in an ideal quantum computer with no error, and if delocalized states do exist, then the real experiment should get the same distribution as the expected. For a classical computers, it is hard to track delocalized states, so its outputs should be uniformly distributed. In real experiments, noise always occurs, and it will add randomness to the system, causing the distribution of the measured results to be more uniform. To conclude, if there is no delocalized state, the experiment distribution should be uniform. If delocalized states appear, the best it can do is to get the same result as the expected. It is not possible that the experiment will still get Porter-Thomas distribution but the dominant bitstrings are significantly different from the simulation, which means the simulation is totally wrong. Therefore, for real expeiment with noise, the results will always lie in between expected and uniform distributions, unable to be better or worse.

We now assume the existence of delocalized states, and set $q(x_i)$ in Eq.(3) to be the expected simulation distrubution P_{expected}. As $p(x_i) = P_{\text{measured}}$ changes from P_{expected} gradually to incoherent uniform distribution $P_{\text{incoherent}} = 1/N$ with dominant bitstrings mostly unchanged, which represents a delocalized state result with increasing error rate, instead of a result that does not agrees with the similation, it is easy to prove that the cross-entropy of these two distributions $S(P_{\text{measured}}, P_{\text{expected}}) = -\sum_{x_i} P_{\text{measured}} \log(P_{\text{expected}})$ is expected to increase (see Appendix A.1). This matches with our understanding of entropy that disordered state has larger entropy. Since P_{measured} cannot be more concentrated than P_{expected}, which makes $S(P_{\text{measured}}, P_{\text{expected}})$ larger than $S(P_{\text{expected}})$, nor worse than P_{expected} that dominant bitstrings do not match with the expected, which can make it smaller than $S(P_{\text{incoherent}}, P_{\text{expected}})$, the minimum and maximum of $S(P_{\text{measured}}, P_{\text{expected}})$ are thus $S(P_{\text{incoherent}}, P_{\text{expected}})$ and $S(P_{\text{expected}})$.

Therefore, the cross-entropy $S(P_{\text{measured}}, P_{\text{expected}})$ can show us whether delocalized states exists. If so, then the less the error rate of the experiment, the closer P_{measured} is to P_{expected}, and the smaller the cross-entropy is. If not, or error too serious, the cross-entropy will be $S(P_{\text{incoherent}}, P_{\text{expected}})$.

Additionally, if the system generates distribution that follows Porter-Thomas distribution, but not due to delocalized states, which means the measured dominant bitstrings will be significantly different from the expected, then according to Appendix 1, the cross-entropy will be even larger than $S(P_{\text{incoherent}}, P_{\text{expected}})$.

To conclude, the cross-entropy $S(P_{\text{measured}}, P_{\text{expected}})$ can be a signature of delocalized states. If $S(P_{\text{measured}}, P_{\text{expected}}) = S(P_{\text{incoherent}}, P_{\text{expected}})$, the existence of delocalized states is not shown. If $S(P_{\text{expected}}) \leq S(P_{\text{measured}}, P_{\text{expected}}) < S(P_{\text{incoherent}}, P_{\text{expected}})$, it demonstrates delocalized states, and the less the cross-entropy is, the less the error rate in the experiment is. And if $S(P_{\text{measured}}, P_{\text{expected}}) > S(P_{\text{incoherent}}, P_{\text{expected}})$, while P_{measured} still follows Porter-Thomas distribution, then it illustrates that there will be a different physical reason to generate such distribution.

3.2 Fidelity and Cross-Entropy Difference

As discussed above, $S(P_{\text{measured}}, P_{\text{expected}})$ can signify delocalized states, thus may also signify computational complexity, but it does not seem so obvious and intuitive as a signature. Regardless of the additional case above related to different physical reasons, delocalized states can be signified by the distance between the measured distribution, the incoherent uniform distribution, and the expected distribution.

The distance between two distribitions $p(x_i)$ and $q(x_i)$ is measured using the KL-divergence D_{KL} [**12**]

$$D_{KL}(p(x_i), q(x_i)) = S(p(x_i), q(x_i)) - S(q(x_i)) \tag{4}$$

One way to show how much the measured distribution resembles the expected distribution is to calculate the difference of the distances $D_{KL}(P_{\text{measured}}, P_{\text{expected}})$ and $D_{KL}(P_{\text{incoherent}}, P_{\text{expected}})$, which is just the cross-entropy difference

$$CED = S(P_{\text{incoherent}}, P_{\text{expected}}) - S(P_{\text{measured}}, P_{\text{expected}}) \tag{5}$$

The cross-entropy difference gets smaller as the experiment distribution diviates from the ideal expected distribution to the incoherent uniform distribution, due to the increasing error rate.

Therefore, the value of the cross-entropy difference can be used to evaluate the error rate of the experiment. To provide a speedup, a quantum system is required to perform a computational task in a large enough space with low enough errors. This is called the fidelity criterion, the second criterion to demonstrate quantum supremacy. Fidelity for the RQCs is therefore introduced by the Google's team, which labels the extent of the measured distribution that overlaps the expected distribution, to represent the accuracy or error rate of the experiment. Fidelity can be quantified using the normalized distance of KL-divergences

$$\mathcal{F} = \frac{D_{KL}(P_{\text{incoherent}}, P_{\text{expected}}) - D_{KL}(P_{\text{measured}}, P_{\text{expected}})}{D_{KL}(P_{\text{incoherent}}, P_{\text{expected}})} \tag{6}$$

The fidelity is normalized. When the distance between the measured and expected probabilities is small, the fidelity approaches 1, while when the measured probability approaches an incoherent mixture, i.e. uniform distribution, the fidelity approaches 0. $P_{\text{incoherent}}$ describes the behavior observed observed after a large num-

ber of cycles, that decoherence reduces the overlap of expected probability until reaching uniform.

A high fidelity will demonstrate low errors in measurement, which is the second criterion in demonstrating quantum supremacy. In order to get high fidelity, we have to first use equipment with less error rate, but this is the engineers' responsibility. Secondly, we have to make the experimental probabilities resembles the expected (exponential) distribution after as few cycles (or depth) as possible, to avoid the influence of decoherence, which is one of our goals in our experiment.

3.3 Cross-Entropy Benchmarking

3.3.1 Cross-Entropy Benchmarking Fidelity

One method of verifying that the quantum processor is working properly is called cross-entropy benchmarking (XEB) [**7**, **10**, **12**], comparing measured probability with the expected distribution with fidelity computed with cross-entropy, just as given in Eq.(6), which can can also be expressed as

$$\mathcal{F}_{\text{XEB}} = \frac{S(P_{\text{incoherent}}, P_{\text{expected}}) - S(P_{\text{measured}}, P_{\text{expected}})}{S(P_{\text{incoherent}}, P_{\text{expected}}) - S(P_{\text{expected}})} \tag{7}$$

To measure the fidelity in real experiment, we may first simplify this formula to a more intuitive version. Recalling the definition of cross-entropy $S(p(x_i), q(x_i)) = -\sum_{x_i} p(x_i) \log q(x_i)$, we find that it is just the weighted average of $-\log q(x_i)$ over all bitstrings x_i with weight being $p(x_i)$.

3.3.2 Calculation of Probability Distribution of Bitstrings

To get the expected probability from ideal quantum circuit, we may simulate it on a quantum simulator for long enough M shots and compute the probability for each bitstring $P_{\text{expected}}(x_i) = \sum_{x_i} /M$. We should also find that the expected probabilities are ought to follow the Porter-Thomas distribution. To prove it, we then need to draw the histogram of the distribution of all the probabilities $P_{\text{expected}}(x_i)$, and see if their probability density distribution follows Porter-Thomas distribution.

For a simulation of an n-qubit system and the dimension of the bitstring basis $N = 2^n$, from which we have calculated $P_{\text{expected}}(x_i)$, we first draw the histogram by probability intervals $P_{\text{expected}}(x_i) \in [p, \Delta p)$, of the number of such bitstrings x_i, which we denote as Nf-p. Then if we divide the count of bitstrings with the dimension $N = 2^n$, the histogram will show the fraction/probability of bitstrings in each interval. After that we devide this fraction/probability with the width of each interval, the result will be the probability density $f(p)$ of bitstrings x_i to get $P_{\text{expected}}(x_i) \in [p, \Delta p)$, labeled as f-p. So now we can compare the simulated probability density distribution with the theoretical Porter-Thomas distribution $f(p) = N\mathrm{e}^{-Np}$, to see if P_{expected} follows PTD. Or we can also multiply p with $N = 2^n$, to get the f-Np histogram. For f-p, the area is the fraction/probability of bitstrings; for Nf-p and f-Np, the area is the number of bitstrings.

For P_{measured}, we will do the same thing above but on a real quantum processor. Then we can get the probabilities of bitstrings P_{measured}, and the probability density distribution $f(P_{\text{measured}})$. If the fidelity to be calculated later is non-zero, by comparing P_{measured} and P_{expected}, the effect of errors on delocalized states can be seen, which tend to flatten P_{measured} towards $P_{\text{incoherent}}$ by prediction.

In experiment, if the system is small, then it is certainly able to observe each bitstring at least once, with large enough amount of shots M. In this circumstance, we can directly divide the count of each bitstring by M to get $P(x_i)$. But for large systems like with $N \geq 2^{20}$, it is hardly possible to observe any bitstring more than once, so the observed result will be some discrete bitstrings. However, we may still expect a higher density of observed bitstrings for those with similar and higher P_{measured} (imagine M is large enough so that we can know it properly, although unrealistic). Therefore, if as predicted, P_{measured} resembles P_{expected}, then when bitstrings are sorted by P_{expected}, we may expect to observe a higher density of the observed where P_{expected} is larger.

In this case, even though it is not possible to really get the exact P_{measured} through experiment, we can still demonstrate if it follows Porter-Thomas distribution by sorting in the way above, and draw the histogram with a width larger than one bitstring. We can see this probability histogram as P_{measured} by assuming the order of sorted observed bitstrings highly resembles the expected. So for sorted bitstrings by P_{expected} (or imagine by P_{measured}), the definition of P_{measured} when observed bitstrings are discrete is instead the fraction of observed bitstrings x_j averaged over a width of $\Delta N > 1$ bitstrings x_i with similar probabilities $P_{\text{measured}}(x_i)$.

Although we have understood the meaning of P_{measured} in large systems, we may not have to directly calculate it and then use it to calculate fidelity in experiment. We will discuss later how to calculate fidelity in this case.

The incoherent distribution can either be measured with long enough cycles of random circuit, or, to be simple, directly use the uniform distribution $P_{\text{uniform}} = \frac{1}{N} = \frac{1}{2^n}$, which we will use in the following discussion.

3.3.3 Calculation of Cross-Entropy Benchmarking Fidelity

Now with P_{expected} and $P_{\text{incoherent}}$ known, it is easy to calculate $S(P_{\text{expected}})$ and $S(P_{\text{incoherent}}, P_{\text{expected}})$. The major work then is to compute $S(P_{\text{measured}}, P_{\text{expected}})$, since P_{measured} is not always obviously available. For small systems, with relatively accurate P_{measured}, of course we can directly use Eq.(7), since all the probabilities are available. However, for large systems that not every bitstring can be observed, we cannot directly copy the definition of P_{measured} of small systems. P_{measured} will have to be evaluated by the average observed probability of $\Delta N > 1$ bitstrings with similar P_{measured}, which is hard to operate. therefore, we want to see whether the direct calculation of P_{measured} is avoidable when computing $\mathcal{F}_{\text{XEB}}$.

We can rewrite the definition of $S(P_{\text{measured}}, P_{\text{expected}})$ by intervals of sorted bitstrings with similar P_{expected}. Assume all M measured bitstrings are $x_{\text{measured},j}$, the width of an interval is ΔN bitstrings, and intervals ar X_k, then we have

$$\begin{aligned}
S(P_{\text{measured}}, P_{\text{expected}}) &= -\sum_{x_i}^{N} P_{\text{measured}}(x_i) \log P_{\text{expected}}(x_i) \\
&= -\sum_{x_i}^{N} \frac{n_{\text{measured}}(x_i)}{M} \log P_{\text{expected}}(x_i) \\
&= -\frac{1}{M} \sum_{X_k}^{N/\Delta N} \sum_{x_i \in X_k}^{\Delta N} n_{\text{measured}}(x_i) \log P_{\text{expected}}(X_k) \\
&= -\frac{1}{M} \sum_{x_{\text{measured},j}}^{M} \log P_{\text{expected}}(x_{\text{measured},j}) \\
&= \langle -\log P_{\text{expected}}(x_i) \rangle_{\text{measured},j}
\end{aligned} \tag{8}$$

It means $S(P_{\text{measured}}, P_{\text{expected}})$ is the average of $-\log P_{\text{expected}}(x_i)$ over all observed bitstrings x_j. Clearly, it is true for either large or small systems, while for the

small, just set $\Delta N = 1$. Since all x_i's in the basis can be observed more than once in small systems, when we group observed x_j's by x_i and devide by M, we simply get the true $P_{\text{measured}}(x_i)$ and then go back to the original definition.

$S(P_{\text{incoherent}}, P_{\text{expected}})$ and $S(P_{\text{expected}})$ may also be calculated as $-\log P_{\text{expected}}(x_i)$ averaged over uniform distribution ($\{x_i\}$ is the simpliest) and all simulated bitstrings. However, $S(P_{\text{incoherent}}, P_{\text{expected}}) = \log N + \gamma$, $S(P_{\text{incoherent}}, P_{\text{expected}}) = \log N - 1 + \gamma$, if we assume $P_{\text{incoherent}} = P_{\text{uniform}}$, and P_{expected} follows Porter-Thomas distribution, as the dimension of bitstring basis $N = 2^n \to +\infty$, where n is the number of qubits, and $\gamma \approx 0.577216$ is the Euler–Mascheroni constant, according to Appendix A.2 [**10**].

For large systems, plug them in Eq.(7), and $\mathcal{F}_{\text{XEB}}$ will be simplified as

$$\mathcal{F}_{\text{XEB}} = \log N + \gamma - \langle -\log P(x_i) \rangle_j \tag{9}$$

where $P(x_i)$ is the expected ideal probability distribution (by simulation). The average is over all M measured bitstrings x_j. Or to be safe, we can also keep $S(P_{\text{incoherent}}, P_{\text{expected}})$ (and $S(P_{\text{expected}})$) for actual calculation instead of plugging in theoretical formulas. This may be essential especially for small systems, since the theoretical values above are valid for $N = 2^n \to +\infty$.

It can be infered from Eq.(9) that, when $P_{\text{measured}} \to P_{\text{incoherent}}$, $\langle -\log P(x_i) \rangle_j \to S(P_{\text{incoherent}}, P_{\text{expected}}) = \log N + \gamma$, $\mathcal{F}_{\text{XEB}} \to 0$. And as it approaches P_{expected}, $\langle -\log P(x_i) \rangle_j \to S(P_{\text{expected}}) = \log N - 1 + \gamma$, $\mathcal{F}_{\text{XEB}} \to 1$. So $\mathcal{F}_{\text{XEB}}$ agrees with our intuition of how much P_{measured} resembles the ideal P_{expect}, or how accurate the actual quantum processor is working. When there is no error, and delocalized states exist, experiment results will follow P_{expect}, and thus produce $\mathcal{F}_{\text{XEB}} = 1$. Sampling from maximally mixed states (complete incoherent), corresponding to uniform distribution, there is no overlap of the sampling and the expected distribution, producing

$\mathcal{F}_{\text{XEB}} = 0$. When there are errors in experiment, the condition should lie in between, producing $0 < \mathcal{F}_{\text{XEB}} < 1$ that correlates with the error rate.

Additionally, if the entropy is defined as $S(P) = -\sum_i P \log_a P$, with base a, then it is easy to see that such entropy is simply $1/\log a$ of Eq.(2) defined above with natural logarithm. Therefore, when plugging such entropy in Eq.(7), all $\log a$ coefficients cancel, thus it will not affect the value of $\mathcal{F}_{\text{XEB}}$.

3.4 Linear Cross-Entropy Benchmarking

Google also computes the fidelity with linear cross-entropy benchmarking (LXEB) [**7**], which is similar to XEB. The definition of LXEB fidelity given in Google's paper for a large n-qubit system, with dimension $N = 2^n$, is

$$\mathcal{F}_{\text{LXEB}} = 2^n \langle P(x_i) \rangle_j - 1 \tag{10}$$

Here $P(x_i)$ is still the ideal probability distribution. The average is over all M observed bitstrings x_j. We will show below that such benchmarking is a similar version of XEB, just replacing the (cross-)entropy with a linear (cross-)entropy $S_{\text{L}}(p)$ ($S_{\text{L}}(p, q)$) defined by

$$S_{\text{L}}(p(x_i)) = \sum_{x_i} p(x_i)(1 - p(x_i)) \tag{11}$$

$$S_{\text{L}}(p(x_i), q(x_i)) = \sum_{x_i} p(x_i)(1 - q(x_i)) \tag{12}$$

Compared with $-\log p$ in the definition of original (von Neumann) entropy Eq.(2), $1 - p$ has the following similar properties, which qualify Eq.(11) and Eq.(12) as an entropy and a cross-entropy in the normal sense.

Firstly,

$$\begin{aligned} -\log p|_{p=1} &= 1-p|_{p=1} = 0 \\ -p\log p|_{p=0} &= p(1-p)|_{p=0} = 0 \end{aligned} \tag{13}$$

Therefore, for completely ordered probability distribution

$$p(x_i) = \begin{cases} 1 & \text{if } x_i = x_{i_0}, \\ 0 & \text{if } x_i \neq x_{i_0}. \end{cases} \tag{14}$$

$S(p(x_i))$ and $S_{\mathrm{L}}(p(x_i))$ both go to 0, which accords with the definition of entropy as the degree of disorder of the system.

Secondly, $-\log p$ and $1-p$ both monotallically increase as p decrease from 1 to 0. It can be infered from this condition, according to Appendix A.3, that $S(p(x_i))$ and $S_{\mathrm{L}}(p(x_i))$ both get higher as the system becomes more disordered until being completely disordered (uniform distribution), which is another property of entropy in the normal sense. Also, when $q(x_i)$ is set unchanged, and $p(x_i)$ evolves from $q(x_i)$ towards uniform distribution, with an increasing degree of disorder, $S(p(x_i), q(x_i))$ and $S_{\mathrm{L}}(p(x_i), q(x_i))$ both in the meanwhile increase. Therefore, $S_{\mathrm{L}}(P_{\mathrm{measured}}, P_{\mathrm{expected}})$ can also be used to show to what extent the measured distribution resembles the expected distribution.

In accordance with Eq.(7), the LXEB fidelity can be defined likewise

$$\mathcal{F}_{\mathrm{LXEB}} = \frac{S_{\mathrm{L}}(P_{\mathrm{incoherent}}, P_{\mathrm{expected}}) - S_{\mathrm{L}}(P_{\mathrm{measured}}, P_{\mathrm{expected}})}{S_{\mathrm{L}}(P_{\mathrm{incoherent}}, P_{\mathrm{expected}}) - S_{\mathrm{L}}(P_{\mathrm{expected}})} \tag{15}$$

Replacing all the $-\log P_{\mathrm{expected}}$ in the equations of XEB with $1-P_{\mathrm{expected}}$, we will get the corresponding results for linear cross-entropy benchmarking. For instance,

$$S_{\mathrm{L}}(P_{\mathrm{measured}}, P_{\mathrm{expected}}) = \langle 1-P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{measured},j} = 1-\langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{measured},j} \tag{16}$$

$S_{\mathrm{L}}(P_{\mathrm{incoherent}}, P_{\mathrm{expected}}) \to 1 - \frac{1}{N}$, $S_{\mathrm{L}}(P_{\mathrm{expected}}) \to 1 - \frac{2}{N}$, when $N = 2^n \to +\infty$, according to Appendix A.4. Plug into Eq.(15), and we will get, for large systems

$$\begin{aligned}\mathcal{F}_{\mathrm{LXEB}} =& \frac{1 - \frac{1}{N} - (1 - \langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{measured},j})}{1 - \frac{1}{N} - (1 - \frac{2}{N})} \\ =& N\langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{measured},j} - 1 \\ =& 2^n\langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{measured},j} - 1\end{aligned} \tag{17}$$

just as Google presents Eq.(10) [**7**].

Like XEB, $\langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{incoherent},j} \xrightarrow{N\to+\infty} \frac{1}{N}$, $\langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{expected},j} \xrightarrow{N\to+\infty} \frac{2}{N}$, so, as $P_{\mathrm{measured}} \to P_{\mathrm{incoherent}}$, $\mathcal{F}_{\mathrm{LXEB}} \to N\langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{incoherent},j} - 1 = 0$, and as $P_{\mathrm{measured}} \to P_{\mathrm{expected}}$, $\mathcal{F}_{\mathrm{LXEB}} \to N\langle P_{\mathrm{expected}}(x_i)\rangle_{\mathrm{expected},j} - 1 = 1$. For small systems, however, we should still use the original definition Eq.(15) to compute $\mathcal{F}_{\mathrm{LXEB}}$, since for small $N = 2^n$, the error of $S_{\mathrm{L}}(P_{\mathrm{expected}})$ is large, which may cause considerable error of $\mathcal{F}_{\mathrm{LXEB}}$.

In conclusion, fidelity can also be calculated in linear cross-entropy benchmarking by Eq.(10) or Eq.(17). Such $\mathcal{F}_{\mathrm{LXEB}}$ works similarly to the $\mathcal{F}_{\mathrm{XEB}}$ in quantifying how properly the actual quantum processor is working.

In section 4, we can use either $\mathcal{F}_{\mathrm{XEB}}$ or $\mathcal{F}_{\mathrm{LXEB}}$, and compute in the methods stated above.

4 Random Circuits Sampling on Qiskit

In this section we will describe our methodology and steps for running random circuit sampling problem on the IBM devices. With the help of the Qiskit tools, we coded some quantum circuits with randomly generated gates, and ran the circuits on a 5-qubit as well as a 15-qubit superconducting quantum computer.We also simulated the circuits classically using Qiskit, such that we can calculate the (linear) cross-entropy benchmarking fidelity and compare the results with Google's [**7**].

4.1 Random Circuits

We use the set $\{CNOT, U := \text{All Unitary Gate}\}$, a universal gate set that can lead to any state in the 2^n-state space to generate random circuits. Given an n-qubits system, we will do the following steps to generate a layer of random circuit:

1) Generate and apply a random unitary gate for every qubits.

2) Depending on the geometry and connectivity of the system (for better fidelity), randomly connect some qubits pairwise by applying a CNOT gate between them.

The randomness on the unitary gates and the two qubit gates breaks the symmetry of the system and should produce delocalized states. We will apply several layers of such random circuit until the signature of the quantum chaos is observed on simulator. Depending on the geometry and the size of the system, the number of layers can be quite different, though vaguely it follows a $n^{1/D}$ relation, for n to be the size of the system and D to be the dimension of the system (a chain or a square or a 3D lattice) [**10**].

4.2 5-Qubits System

Fig. 1 below shows the geometry, connectivity, and the error rate for the IBM-QX2 5-qubit superconducting quantum computer [**13**]. As qubit 2 is connected to all the other qubits, while qubit 0, 1 and qubit 3, 4 are connected pairwise, therefore, we used the following scheme for step 2): Connect qubit 0 & 1, qubit 3 & 4, qubit 2 & a randomly selected qubit, in each layer. We did simulation based on such random circuit, and observed Porter-Thomas distribution after 5 layers. Thus we generated 5 layers of such random circuit, shown by Fig. 2.

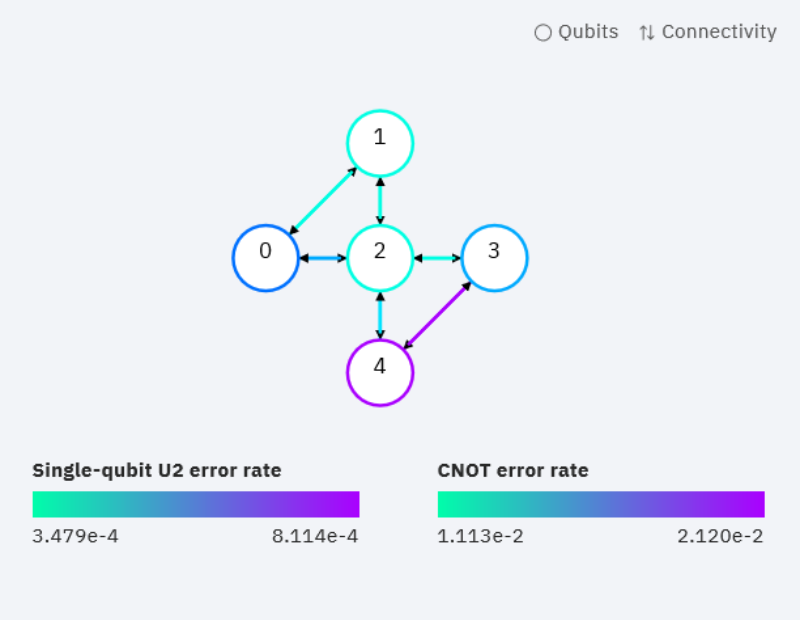

Fig. 1: *Hardware description of the geometry, connectivity, and the error rate for IBM QX2 architecture.*

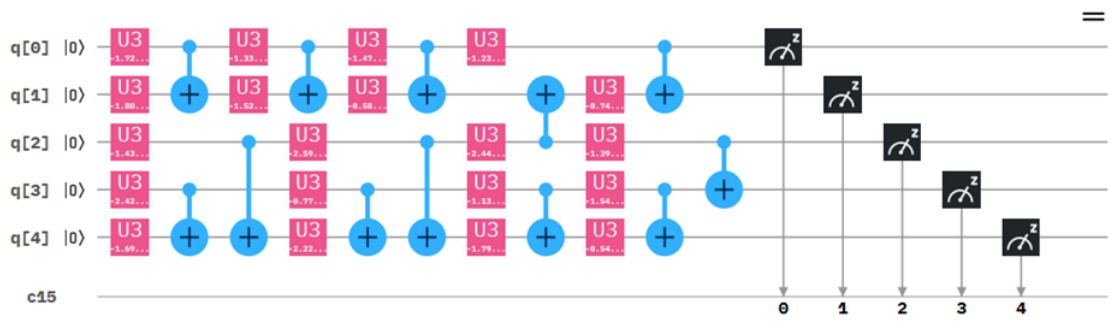

Fig. 2: *Diagram of the 5-qubit random quantum circuit used in our experiment.*

4.3 15-Qubit System

Fig. 3 below shows the geometry, connectivity and error rate for the IBM-Q-16 Melbourne 15-qubit quantum computer [**13**]. We tried to run a 15-qubit circuit on it, but due to the limitation on the number of shots one can get for every experiment, we could not get enough data to explore the huge 2^{15} Hilbert space. Thus we tried a 10-qubit circuit (using qubits 2-6,8-12) instead. Based on its rectangular geometry, we used the following scheme for step 2): Randomly pick 3-5 qubits and randomly connect them to a nearest neighbor qubit (either horizontal or vertical neighbor) that is not connected. Otherwise keep such qubit alone. We did simulation based on such random circuit, and we observed Porter-Thomas distribution after 40 layers. Thus we generated 40 layers of such random circuit, where the first part of the circuits is shown by Fig. 4.

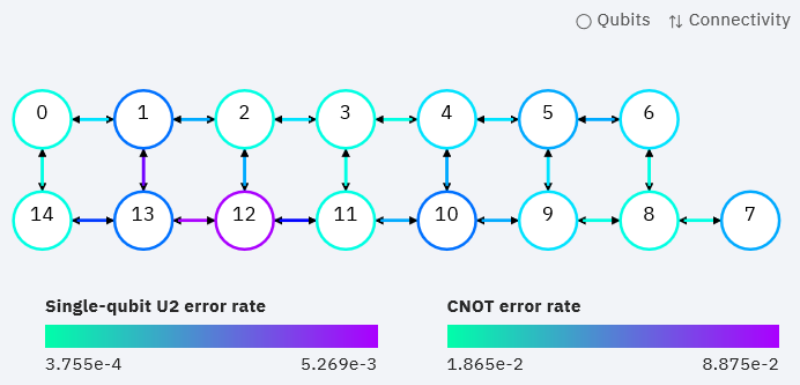

Fig. 3: *Hardware description of the geometry, connectivity, and the error rate for IBM-Q-16-Melbourne architecture.*

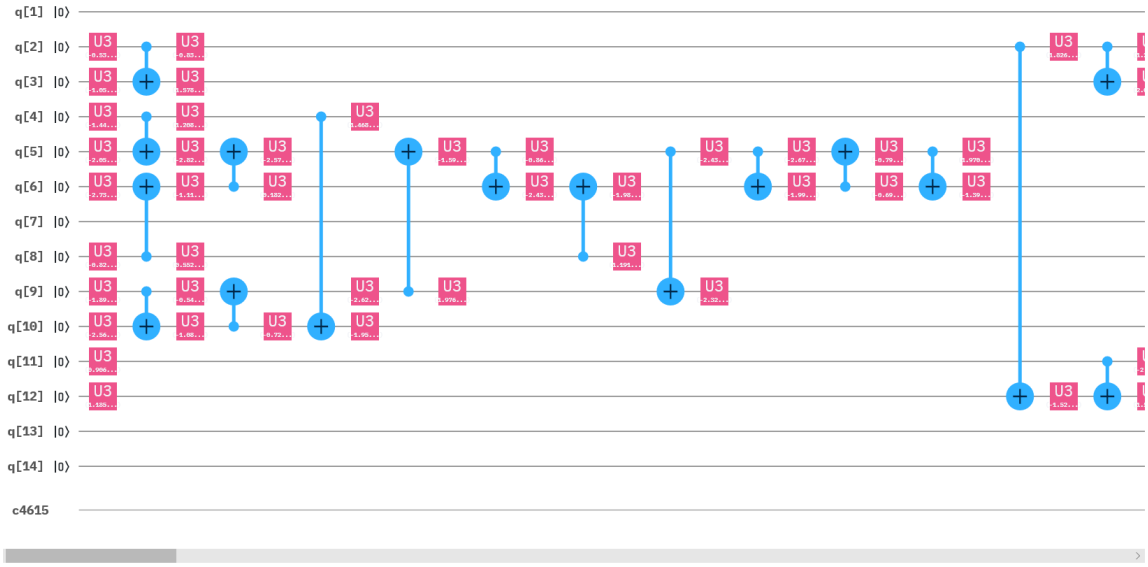

Fig. 4: *Diagram of the 10-qubit random quantum circuit used in our experiment.*

5 Results

5.1 5-Qubits Circuits

Fig. 5 is the probability density histogram for the 5-qubit circuit.

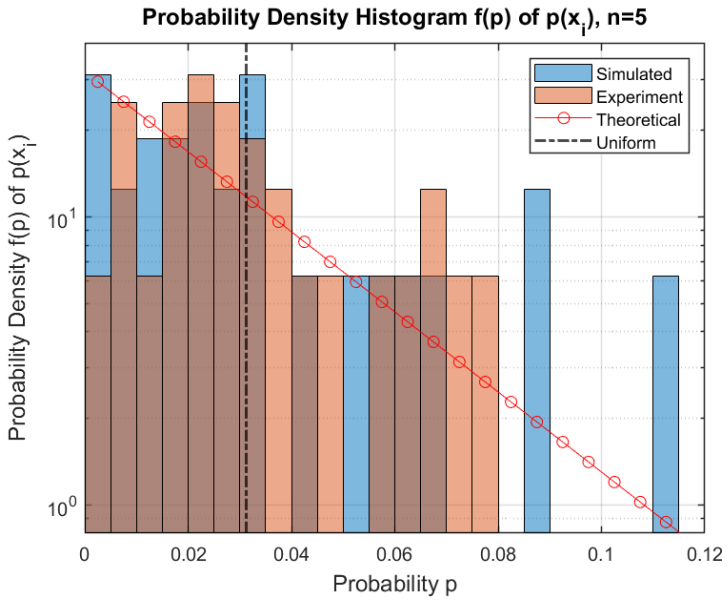

Fig. 5: *Probability density histogram, for a 5-qubit random circuit, of the probabilities $p(x_i)$ of all $N = 2^5$ bitstrings in the Hilbert space. The Porter-Thomas distribution (PTD) and uniform distribution are plotted in the same figure to help compare how much the results resembles PTD. As we can see, although not perfect, both the simulation and experiment results mainly follow PTD. Since we know from Fig. 1 that the average error rate is about $r \sim 10^{-3}$, according to Fig. 8, our experiment result resembles the behavior of simulated $f(p)$ with Pauli error rate of such scale [**10**], as the peak of probability distribution moves from $p = 0$ of the (ideal) simulation, to $p \approx 0.02$ of actual experiment. Therefore, the effect of errors on random circuit sampling can be shown from our result.*

The histogram shows the quantum-chaotic nature of the states, as it scales

exponentially, behaving similarly to the Porters-Thomas Distribution. It does not strictly follow the distribution as the small system size limits us from sampling more from the distribution. A better fit of simulated distribution can be found for 10-qubit circuits as we can sample more. Also, as for the 5-qubit circuit, we utilize the geometry and error information. This results in a small circuit (only 5 layers) and low error rate hardware, making our experimental data quite close to the simulated result.

Since we know from Fig. 1 that the average error rate of IBM-QX2 device is at the scale $r \sim 10^{-3}$, according to Fig. 8, our experiment result resembles the behavior of simulated $f(p)$ of a larger system with such Pauli error rate [**10**], as the peak of the probability moves from $p = 0$ of the (ideal) simulation, to $p \approx 0.02$ of actual experiment. Therefore, the effect of errors on random circuit sampling can be qualitatively shown from our result.

Another prospective of the quantum-chaotic nature can be shown by the log-scale plot of the probability amplitudes of every state x_i in the $N = 2^n$ Hilbert space sorted by the simulated probabilities, shown by Fig. 6. One can see that the simulated result matches well with the theoretical prediction. One can also have a better look on how the experimental data differs from the simulated data in this perspective. The experimental result matches better for the states with higher probability amplitudes and suffer from matching states with small probability amplitudes. This generally holds for most of the "quantum-computers" in this era, as noise makes the distribution converges to uniform distribution. It can be found in google's paper that, for a 52-qubit system, the results are almost uniform [**7**].

As the figure shows, the probability of least probable state ($i = N = 2^5$) scales quite perfectly as the discrete Porter-Thomas distribution Eq.(39) predicts, unlike

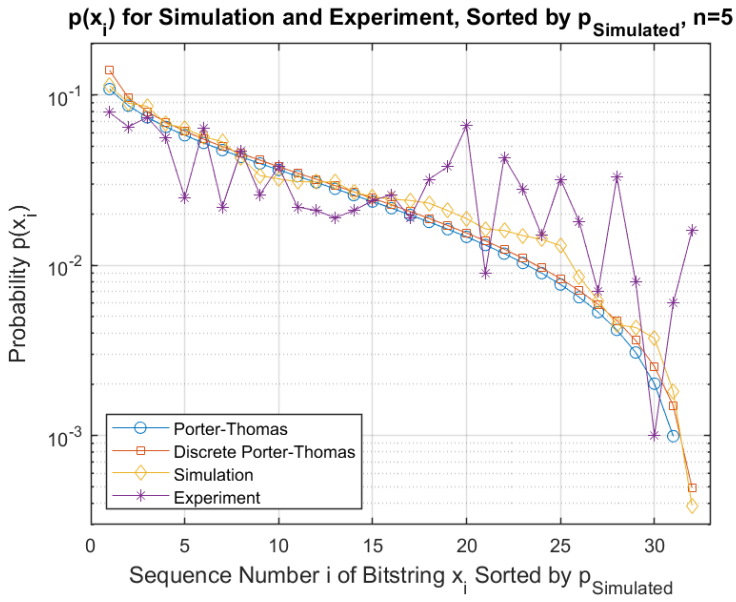

Fig. 6: *Simulated and measured probabilities of every bitstring x_i in $N = 2^5$ Hilbert space, sorted in the decreasing order of simulated probabilities. The simulation result matches well with the theoretical prediction for the half number of bitstrings with higher probability amplitudes, and so is the experiment result. Both of them suffer from matching low probability states, due to error, and it will lead results to uniform distribution.*

directly using Eq.(37), which gives out absolutely 0.

Fig. 6 is already good enough to prove the exiatance of delocalized states as the dominant bitstrings all matches nearly perfectly. But to quantitively indicate it, and also show how accurate the chosen quantum circuit works, we then calculate the fidelities defined in different benchmarking, as discribed in section 3 and Appendix A.4, show their similarities and their relation to the error rate.

Let's start with XEB. Since the system is small, we have already known the expected (simulated) probability of each bitstring, and the corresponding experimental

probability of the same bitstring. To calculate the fidelity in different benchmarkings, we can first directly calculate the (cross-)entropy by Eq.(2) & Eq.(3), or by Eq.(8), and plug them into Eq.(7), then calculate another $\mathcal{F}_{\text{XEB}}$ in the simplified Eq.(9) that works better for large systems and compare them. $\mathcal{F}_{\text{XEB}}$ calculated from original Eq.(7)

$$\mathcal{F}_{\text{XEB,5,ori}} = 0.5741 \tag{18}$$

while from the simplified Eq.(9) is

$$\mathcal{F}_{\text{XEB,5,sim}} = 0.5788 \tag{19}$$

with the constants calculated from simulated and experimental probabilities by the corresponding equations of Eq.(47) and Eq.(48) being

$$\begin{aligned} F_{1,5} &= \log 2^5 - \sum_{i=1}^{2^5} \frac{1}{2^5} \left[-\log P_{\text{simulated}}(x_i) \right] = -0.4884 \\ F_{2,5} &= \log 2^5 - \sum_{i=1}^{2^5} P_{\text{simulated}}(x_i) \left[-\log P_{\text{simulated}}(x_i) \right] = 0.3500 \\ F_{2,5} - F_{1,5} &= 0.8383 \end{aligned} \tag{20}$$

Referring to Table 1, $F_{1,5}$ and $F_{2,5}$ diviate quite a bit from the theoretical series results for $n = 5$, and actually match with those for about $n = 2$'s. This indicates the error that occur in both our simulation and actual experiment. However, the results of the two $\mathcal{F}_{\text{XEB}}$'s are so much the same. It is not hard to understand, since after all, the error rate in the 5-qubit experiment of $r \sim 10^{-3}$ is rather satiesfactory.

Then we can calculate $\mathcal{F}_{\text{LXEB}}$, also both from the original definition Eq.(15)

$$\mathcal{F}_{\text{LXEB,5,ori}} = 0.5649 \tag{21}$$

and the simplified version Eq.(17) valid for large systems

$$\mathcal{F}_{\mathrm{LXEB},5,\mathrm{sim}} = 0.4311 \tag{22}$$

as well as the corresponding constant F_3 calculated from real data for modification by Eq.(60)

$$F_{3,10} = 2^5 \sum_{i=1}^{2^5} P_{\mathrm{simulated}}(x_i) \cdot P_{\mathrm{simulated}}(x_i) = 1.7632 \tag{23}$$

We can see that the $\mathcal{F}_{\mathrm{LXEB}}$ from the original definition matches perfectly with XEB. While if we directly use the formula for large systems, we will get a bad estimation of $\mathcal{F}_{\mathrm{LXEB}}$, since F_3 diviates seriously from the theoretical series value in Table 2 for $n = 5$ that is about 2, but instead close to that for $n = 2$, like XEB.

Therefore, both $\mathcal{F}_{\mathrm{XEB}}$ and $\mathcal{F}_{\mathrm{LXEB}}$ can illustrate that the error in our simulation and actual measurement worsens our result from $n = 5$ to $n = 2$ if we want to use the theoretical values of the constants F_1, F_2, F_3 from Table 1 and 2.

And according to the generalization at the end of Appandix A.4, we can also define other k'th order (cross-) entropies, while LXEB is just the case for $k = 1$. Now we can have a brief look of the several results of $\mathcal{F}_{\mathrm{kXEB}}$ from such cross-entropies $S_k(p, q) = \sum_i p \cdot (1 - q^k)$ (the following results are by original formulas resembling Eq.(7) and Eq.(15)). For $k = 2, 3, 4, 5, 6$, we have $\mathcal{F}_{\mathrm{kXEB}}$=0.5750, 0.5857, 0.5914, 0.5931, and 0.5927, which are so much the same. So the $\mathcal{F}_{\mathrm{kXEB}}$ might also work as the quantification of the accuracy of the experiment.

With such big non-zero fidelity, it is obvious that the existance of delocalized states can be confirmed. While before we reach a conclusion, let's first see the results for larger systems.

5.2 10-Qubit Circuit

Fig. 7 is the probability density histogram for the 10-qubit circuit.

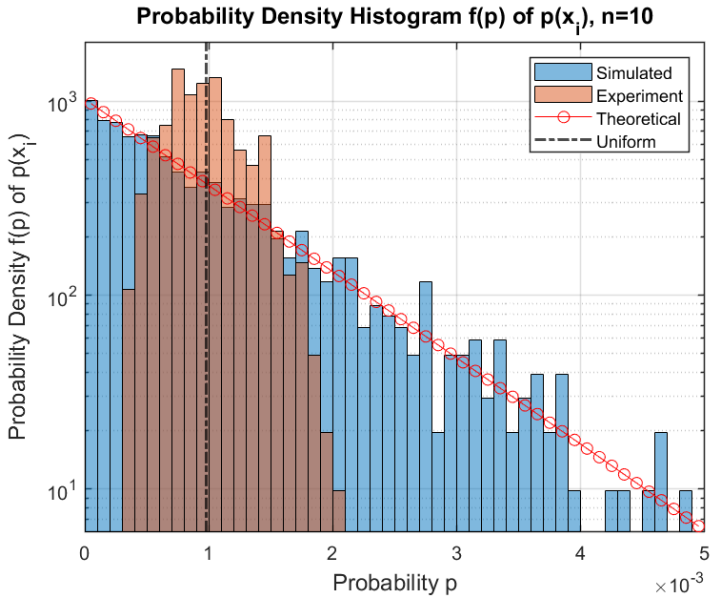

Fig. 7: *Probability density histogram, for a 10-qubit random circuit, of the probabilities $p(x_i)$ of all $N = 2^{10}$ bitstrings in the Hilbert space. The Porter-Thomas distribution (PTD) and uniform distribution are plotted in the same figure to help compare how much the results resembles PTD. As we can see, the simulation result nearly perfectly follows PTD, while the experimental one acumulates arround $p = \frac{1}{N}$. Since we know from Fig. 3 that the average error rate is about $r \sim 10^{-2}$, according to Fig. 8, our experiment result resembles the behavior of simulated $f(p)$ with Pauli error rate of such scale [**10**], with the histogram being a sharp triangular peak about uniform distribution. Therefore, the effect of errors on random circuit sampling can be shown from our result.*

The simulated data follows perfectly the Porter Thomas distribution, and the quantum-chaotic nature of the states,can be shown from the broaden of experimental probabilities from the unoform probability. However, as the geometry of the 10-

qubit system and the error rate is not ideal for running a large random circuit, the experimental data does not look as satiesfactory as that of 5-qubit system. From Figure 7, we can expect a low value for fidelity already. Nevertheless, given the error rate of the IBM system (as high as $r \sim 10^{-2} - 10^{-1}$), the histogram of the experiment data matches with the shape of probability histogram of google's simulation of an $n = 20$ system, by applying such Pauli error rate, shown by Fig. 8 [**10**], with the histogram being a sharp triangular peak about uniform distribution. Therefore, the effect of errors on RQCs can be qualitatively shown from our 10-qubit result.

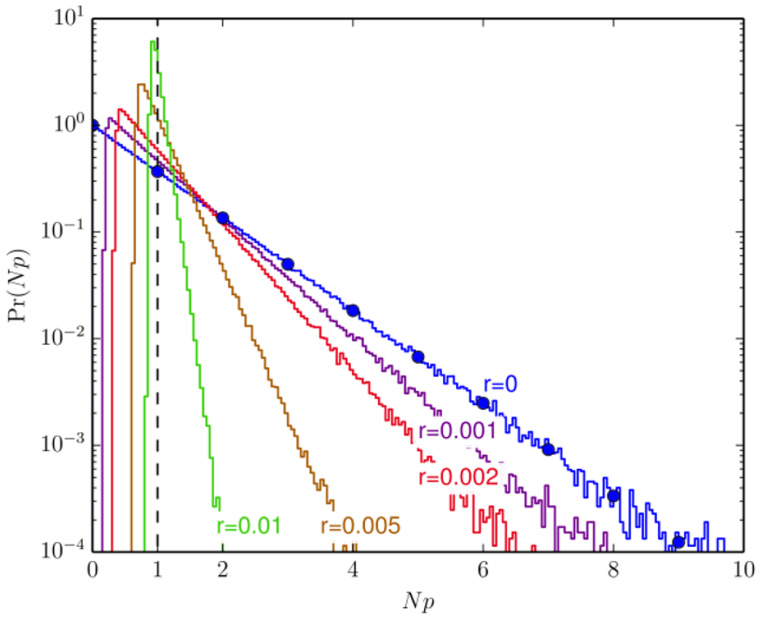

Fig. 8: *The rescaled probability distribution of Google's simulation of random circuit for different Pauli error rates [**10**].* $\Pr(Np) = \mathrm{e}^{-Np}$*, which according to section 3, compared with the normal definition of Porter-Thomas distribution (PTD)* $f(p) = N\mathrm{e}^{-Np} = \frac{1}{N}\Pr(Np)$*, should be an* $\frac{\mathrm{f}}{\mathrm{N}} - \mathrm{Np}$ *plot. The black dashed line correspond to uniform distribution. The blue curve of ideal circuit is very close to PTD (blue dots).*

Fig. 9 shows the log-scale plot of the sorted probability amplitudes of every state x_i in $N = 2^{10}$ the Hilbert space. One can see that the simulated results matches particularly well with the theoretical prediction, while the experimental results almost converge to the uniform distribution, implying a low fidelity. It is hard to demonstrate delocalized states merely from Fig. 9, so we need the help of fidelity.

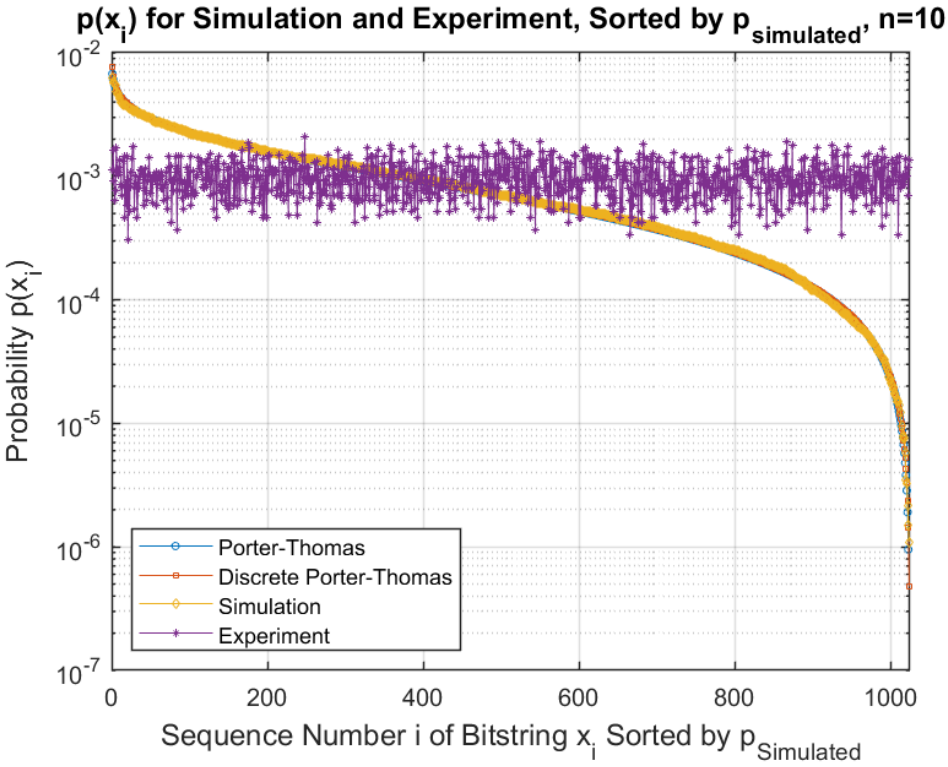

Fig. 9: *Simulated and measured probabilities of every bitstring x_i in $N = 2^{10}$ Hilbert space, sorted in the decreasing order of simulated probabilities. The simulation result matches excellently with the theoretical prediction, while the experiment result nearly converges to uniform distribution $p = \frac{1}{N} \approx 10^{-3}$. Therefore, it is hard to demonstrate delocalized states merely from this plot.*

Let's also start with XEB. Since the system is small, we have already known the $P_{\text{expected}} = P_{\text{simulated}}$ and $P_{\text{experiment}}$ of each bitstring. Following the procedure in section 5.1, the $\mathcal{F}_{\text{XEB}}$ calculated from original Eq.(7) and the simplified Eq.(9) are

$$\begin{aligned} \mathcal{F}_{\text{XEB,10,ori}} &= 3.703 \times 10^{-5} \\ \mathcal{F}_{\text{XEB,10,sim}} &= 1.135 \times 10^{-2} \end{aligned} \tag{24}$$

with the constants F_1 and F_2 calculated from $P_{\text{simulated}}$ $P_{\text{experiment}}$ by the corresponding equations of Eq.(47) and Eq.(48) being

$$\begin{aligned} F_{1,10} &= \log 2^{10} - \sum_{i=1}^{2^{10}} \frac{1}{2^{10}} \left[-\log P_{\text{simulated}}(x_i) \right] = -0.5659 \\ F_{2,10} &= \log 2^{10} - \sum_{i=1}^{2^{10}} P_{\text{simulated}}(x_i) \left[-\log P_{\text{simulated}}(x_i) \right] = 0.4065 \\ F_2 - F_1 &= 0.9724 \end{aligned} \tag{25}$$

Referring to Table 1, although $F_{1,10}$ and $F_{2,10}$ does not perfectly match the theoretical series results for $n = 10$, but actually match with those for about $n = 5$'s, they are still good enough to allow the simplified Eq.(17) to have a relative error of $\sim 10^{-2}$. This indicates the high accuracy of our simulation although error worsens the estimate by a little bit. However, the results of the two $\mathcal{F}_{\text{XEB}}$'s look far different from each other, but actually it can be infered from the nearly uniform distribution and some extent of randomness of our experimental distribution, and the error rate in the 10-qubit experiment of $r \sim 10^{-3}$ or higher. With better devices of lower error rate, maybe we can then see that these two values could match approximately; or by running in fewer layers, even in that case the simulated distribution would not look as perfect as 40 layers' but might significantly increase the accuracy of the actual measurement, and thus expect a higher fidelity.

Then we calculate $\mathcal{F}_{\text{LXEB}}$ from the original Eq.(15) and the simplified Eq.(17)

$$\begin{aligned} \mathcal{F}_{\text{LXEB,10,ori}} &= 2.134 \times 10^{-3} \\ \mathcal{F}_{\text{LXEB,10,sim}} &= 1.985 \times 10^{-3} \end{aligned} \tag{26}$$

and the corresponding F_3 calculated from real data for modification by Eq.(60)

$$F_{3,10} = 2^{10} \sum_{i=1}^{2^{10}} P_{\text{simulated}}(x_i) \cdot P_{\text{simulated}}(x_i) = 1.9299 \tag{27}$$

We can see that these $\mathcal{F}_{\text{LXEB}}$ match with the scale of that by XEB. And with an $F_3 \approx 2$, although not as close to 2 as the ideal expectation in Table 2, but close to that for $n = 5$, the simplified formula Eq.(17) is already an acceptable estimate for $\mathcal{F}_{\text{LXEB}}$. For our data, the relative error of the simplified $\mathcal{F}_{\text{LXEB}}$ is 7%

Therefore, the LXEB results illustrate that the error in our simulation and actual measurement worsens our result from $n = 10$ to $n = 5$ if we want to use the theoretical values of the constant F_3 from Table 2, similar to the effect in the 5-qubit results.

Now for the $n = 10$ experiment, we also calculate several results of $\mathcal{F}_{\text{kXEB}}$ from k'th order cross-entropies $S_k(p, q) = \sum_i p \cdot (1 - q^k)$ (by original formulas resembling Eq.(7) and Eq.(15)), with the same k's as 5-qubit experiment for comparison. For $k = 2, 3, 4, 5, 6$, we have $\mathcal{F}_{\text{kXEB}}$=$4.783 \times 10^{-3}$, 9.776×10^{-3}, 1.491×10^{-2}, 1.907×10^{-2}, 2.206×10^{-2}, which are of similar scales. Together with the results for $n = 5$, we can state the $\mathcal{F}_{\text{kXEB}}$ is qualified to work as the quantification of the accuracy of the experiment, although the exact relationship between $\mathcal{F}_{\text{kXEB}}$ and the accuracy or error rate depends on the value of k.

Therefore $n = 10$, with the non-zero fidelity calcilated in many different benchmarkings, the existance of delocalized states can be confirmed.

5.3 Conclusion

The simulation results for both 5-qubit and 10-qubit random circuits match with Porter-Thomas distribution. And the experiment result for the 5-qubit circuit matches quite well with the simulation result, though for the 10-qubit circuit, the experimental results is limited by the geometry and error rate of the system. Compare to Google's results [**7**], we achieve a fidelity similar to the that expected by extrapolation of Fig. 10, for the 5-qubit circuit using the IBM-QX2 5-qubit system, and a much lower fidelity, for a 10-qubit circuit using the IBM-Q-16-Melbourne 15-qubit system. We believe that we can achieve better results with a system with lower error rate or a better way to utilize the geometry, and we would love to test on a larger system when a suitable system is available. Lastly, we are certainly impressed by the conveniences Qiskit has brought us to perform experiment remotely and instantly, and would love to thank IBM for providing us access to some of quantum systems.

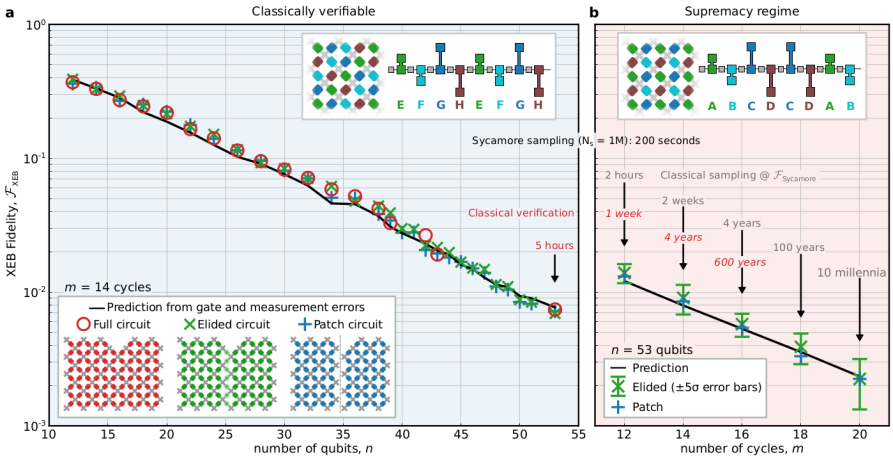

Fig. 10: *Verification of benchmarking methods, in Google's results declaring quantum supremacy, showing their cross-entropy benchmarking fidelity with respect to the number of qubits. For $n = 5$, the extrapolation value of fidelity in this figure is about $0.5 \sim 0.6$.*

A Appendices

A.1 Relation of Cross-Entropy to Error Rate

To prove $S(P_{\text{measured}}, P_{\text{expected}}) = -\sum_{x_i} P_{\text{measured}} \log(P_{\text{expected}})$ will decrease as P_{measured} changes from the expected gradually to the uniform distribution with dominant bitstrings mostly unchanged, which represents a delocalized state result with increasing error rate, instead of a result that does not agrees with the similation, we first sort the bitstrings so that in the decreasing order of P_{expected}.

Since we notice that $\sum_{x_i} P_{\text{measured}} = 1$, $-\sum_{x_i} P_{\text{measured}} \log(P_{\text{expected}})$ can be thought of as a weighted average $\langle -\log(P_{\text{expected}})\rangle_{\text{measured}}$, where $-\log(P_{\text{expected}})$ is thus non-negative and in the increasing order. Now P_{measured} develop gradually from P_{expected} to incoherent uniform distribution $P_{\text{incoherent}} = 1/N$, maybe with noise so that P_{measured} may not be strictly in decreasing order, but instead with small purturbation. We will expect $S(P_{\text{measured}}, P_{\text{expected}})$ to increase, because the weight for smaller $-\log(P_{\text{expected}})$, i.e. for larger P_{expected}, mainly decrease, while on contrast the weight for larger $-\log(P_{\text{expected}})$ mainly increase. This agrees with our understanding of entropy that disordered state has larger entropy.

To mention, if the experiment generates distribution following the same distribution but with different physical reason, which means the dominant bitstrings does not agrees with that of expected delocalized state result, then as we stated above, in this case, $S(P_{\text{measured}}, P_{\text{expected}})$ is even smaller than $S(P_{\text{incoherent}}, P_{\text{expected}})$.

A.2 (Cross-)Entropy of Porter-Thomas Distribution

In this part we will calculate the theoretical value of $S(P_{\text{incoherent}}, P_{\text{expected}})$ and $S(P_{\text{expected}})$, in the limit of $N = 2^n \to \infty$, as well as the probabilities P_{expected} of $N = 2^n$ bitstrings that follow Porter-Thomas distribution. These can be used to simplify Eq.(7) and verify when this simplification is valid.

$P_{\text{incoherent}}(x_i) = P_{\text{uniform}}(x_i)$, so $S(P_{\text{incoherent}}, P_{\text{expected}}) = -\sum_{x_i} \frac{1}{N} \log P_{\text{expected}}$. Now instead of summing over all bitstrings x_i, we sum over all probabilities. By dividing the probabilities from 0 to 1 into intervals p_j with width Δp, we make the histogram of the probabilities, as how we demonstrate Porter-Thomas distribution. And correspondingly, the sum for cross-entropy is also grouped by p_j

$$S(P_{\text{incoherent}}, P_{\text{expected}}) = -\sum_{p_j} \left(\sum_{P_{\text{ex}}(x_i) \in p_j} \frac{1}{N} \log P_{\text{expected}}(x_i) \right) \tag{28}$$

Assume P_{expected} follows Porter-Thomas distribution $f(p) = N\mathrm{e}^{-Np}$, then for an interval p_j, the fraction of bitstrings that $P_{\text{expected}}(x_i) \in p_j$ is $f(p_j)\Delta p$, and the number of such bitstrings is

$$\Delta N_j = N f(p_j) \Delta p = N^2 \mathrm{e}^{-N p_j} \Delta p \tag{29}$$

Plug into Eq.(28), and then we have

$$\begin{aligned} S(P_{\text{incoherent}}, P_{\text{expected}}) &= -\sum_{p_j} \left(\Delta N_j \frac{1}{N} \log p_j \right) \\ &= -\sum_{p_j} \left(N\mathrm{e}^{-N p_j} \log p_j \Delta p \right) \end{aligned} \tag{30}$$

In the limit of $\Delta p \to 0$, the sum can be rewriten as integral

$$\begin{aligned} S(P_{\text{incoherent}}, P_{\text{expected}}) &= -\int_0^1 N\mathrm{e}^{-Np}\log p\mathrm{d}p \\ &= -\int_0^1 \mathrm{e}^{-Np}[\log(Np) - \log N]\mathrm{d}(Np) \\ &= \log N \int_0^N \mathrm{e}^{-x}\mathrm{d}x - \int_0^N \log x\mathrm{e}^{-x}\mathrm{d}x \end{aligned} \tag{31}$$

As $N = 2^n \to +\infty$,

$$\begin{aligned} \lim_{N\to+\infty} S(P_{\text{incoherent}}, P_{\text{expected}}) &= \log N \int_0^{+\infty} \mathrm{e}^{-x}\mathrm{d}x - \int_0^{+\infty} \log x\mathrm{e}^{-x}\mathrm{d}x \\ &= \log N + \gamma \end{aligned} \tag{32}$$

Here $\gamma \approx 0.577216$ is the Euler–Mascheroni constant.

Similarly,

$$\begin{aligned} S(P_{\text{expected}}) &= -\sum_{p_j}\left(\sum_{P(x_i)\in p_j} P_{\text{expected}}(x_i)\log P_{\text{expected}}(x_i)\right) \\ &= -\sum_{p_j}\left(N^2\mathrm{e}^{-Np_j}\Delta p\cdot p_j\log p_j\right) \\ &= -\int_0^1 pN^2\mathrm{e}^{-Np}\log p\mathrm{d}p \\ &= -\int_0^1 Np\cdot\mathrm{e}^{-Np}[\log(Np) - \log N]\mathrm{d}(Np) \\ &= \log N\int_0^N x\mathrm{e}^{-x}\mathrm{d}x - \int_0^N x\log x\mathrm{e}^{-x}\mathrm{d}x \end{aligned} \tag{33}$$

And for large system limit $N = 2^n \to +\infty$,

$$\begin{aligned} \lim_{N\to+\infty} S(P_{\text{expected}}) &= \log N\int_0^{+\infty} x\mathrm{e}^{-x}\mathrm{d}x - \int_0^{+\infty} x\log x\mathrm{e}^{-x}\mathrm{d}x \\ &= \log N + \gamma - 1 \end{aligned} \tag{34}$$

Next we will calculate the probability function $p(x_i)$ of sorted bitstrings x_i, whose probability distribution $f(p(x_i))$ follows Porter-Thomas distribution. We have already known from Eq.(29) the number of bitstrings ΔN_j whose probability is in the vicinity of p_j. In the limit of $\Delta p \to 0$, it becomes a differential equation

$$\begin{aligned} \mathrm{d}x &= -Nf(p)\mathrm{d}p = N^2\mathrm{e}^{-Np}\mathrm{d}p \\ \frac{\mathrm{d}x}{\mathrm{d}p} &= -N^2\mathrm{e}^{-Np} \end{aligned} \tag{35}$$

Here x is the sequence number of sorted bitstrings, in the decreasing order of $p(x_i)$. There is a minus sign since in this order, as p increases, x decreases. The solution is

$$x(p) = N\mathrm{e}^{-Np} \tag{36}$$

with the condition $\int_0^1 x(p)\mathrm{d}p = \int_0^\infty p(x)\mathrm{d}x = 1$, which looks like Porter-Thomas distribution, but with a diffrent meaning. This $x(p) = N\mathrm{e}^{-Np}$ describes the sequence number of bitstring with probability p, simply accidentally being the same as the probability density of such probability, due to its exponential form.

The inverse function is

$$P_{\text{expected}} = P_{\text{PTD}} = p(x) = -\frac{1}{N}\log\frac{x}{N} = \frac{1}{N}\log\frac{N}{x} \tag{37}$$

This is the probability (density) as a function of sequence number of sorted bitstring, which we can draw from experiment and simulation data to demonstrate PTD.

Eq.(37) is the continuous form of the probability distribution, which is more accurate for discrete x for large $N = 2^n$. When $x = N$, i.e. the bitstring with the smallest probability, if we directly use Eq.(37), we will get $P_{\text{expected}}(N) = 0$, apparently non-intuitive, and it will also cause divergence when later calculating some series, which does not make sense. Therefore, expecially for small systems, we may use an integral $\int_{x-1}^{x} p(x)\mathrm{d}x$ as the probability for discrete x.

Since Eq.(37) is for continuous x, it actually represents probability density, so it does not harm to keep the tail with $p > 1$. As we notice

$$\int_0^N p(x)\mathrm{d}x = \int_0^N \frac{1}{N}\log\frac{N}{x}\mathrm{d}x = 1 \tag{38}$$

for contonuous $x \in (0, N]$ and the integral is equal to the total probability 1. So, for discrete i, especially for small systems, we may use the following integral Eq.(39) for each i, instead of directly using Eq.(37), which as discussed will cause troubles.

$$\begin{aligned} P_{\text{PTD,discrete}} = p_i =& \int_{i-1}^{i} p(x)\mathrm{d}x = \int_{i-1}^{i} \frac{1}{N}\log\frac{N}{x}\mathrm{d}x = \frac{x}{N}\left(1-\log\frac{x}{N}\right)\Big|_{i-1}^{i} \\ =& \frac{1}{N}\left\{\log N + 1 - [i\log i - (i-1)\log(i-1)]\right\} \\ =& \frac{1}{N}\left[\log N - \log i + 1 - (i-1)\log\frac{i}{i-1}\right],\ (i = 1, ..., N) \end{aligned} \tag{39}$$

Now we check again if there is still any problem. Compared with from Eq.(37), where we should have $p_i = \frac{1}{N}(\log N - \log i) \geq 0$, there are some additional terms in Eq.(39), $\Delta p_i = \frac{1}{N}\left[1 - (i-1)\log\frac{i}{i-1}\right] \in (0, \frac{1}{N}]$, which are strictly positive. Therefore, the new Eq.(39) is still monotonically decreasing, and look logarithmic, but each term is strictly positive. Moreover, the discrete sum of Eq.(39) is

$$\begin{aligned} \sum_{i=1}^{N} p_i =& \log N - \frac{1}{N}\sum_{i=1}^{N}\log i + 1 - \frac{1}{N}\sum_{i=1}^{N}\left[(i-1)\log\frac{i}{i-1}\right] \\ =& \log N - \frac{1}{N}\sum_{i=1}^{N}\log i + 1 + \frac{1}{N}\sum_{i=1}^{N}\log i - \frac{1}{N}\sum_{i=1}^{N} i\log i + \frac{1}{N}\sum_{i=1}^{N}[(i-1)\log(i-1)] \\ =& \log N + 1 - \frac{i\log i}{N}\Big|_{i=N} + \frac{i\log i}{N}\Big|_{i=0} = 1 \end{aligned} \tag{40}$$

The sum is strictly equal to 1, so all the properties above qualify that Eq.(39) works excellently as the probability distribution for discrete bitstrings x_i.

In the limit of $N = 2^n \to \infty$, when $i \to 1$

$$\Delta p_i = \frac{1}{N}\left[1 - (i-1)\log\frac{i}{i-1}\right] \to \frac{1}{N}\left[1 + (i-1)\log(i-1)\right] \to \frac{1}{N}\left[1 + (i-1)\right] = \frac{i}{N} \tag{41}$$

$$\frac{\Delta p_i}{p_i} \to \frac{i}{\log\frac{N}{i}} \to \frac{i}{\log N} \to 0 \tag{42}$$

When $i \to N \to \infty$

$$\Delta p_i = \frac{1}{N}\left[1 - (i-1)\log\frac{i}{i-1}\right] \to \frac{1}{N}\cdot\frac{1}{2i} = \frac{1}{2Ni} \tag{43}$$

$$\frac{\Delta p_i}{p_i} \to \frac{1}{2i\log\frac{N}{i}} = \frac{1}{2i(\log N - \log i)} \to \frac{1}{2i(N-i)\frac{1}{i}} = \frac{1}{2(N-i)} \tag{44}$$

Therefore, as long as i is not extremely close to N, something like $i \geq N - 10$, the additional terms are always negligible. And why when in such cases Δp_i cannot be omitted is that for Eq.(37), p_i can reach absolutely to 0 when $i = N$, instead of close to 0, in which case we actually do not want to neglect the additional term. That is because making every p_i positive is one of the primary goals of our works above, to avoid divergence by $p_N = 0$ when calculating some series later, to check the validity of Eq.(32), Eq.(34), and then Eq.(9) for small systems.

In conclusion, Eq.(39)→Eq.(37) when $N \to +\infty$, so the continuous version of probabilities following Porter-Thomas distribution is valid for large systems, which further proves the accuracy of the integrals we did at the beginning part of this section, thus the accuracy of Eq.(32), Eq.(34), and Eq.(9) for large systems.

With the expression Eq.(39) of probabilities for discrete i, we can compute the theoretical expectation values of $S(P_{\text{incoherent}}, P_{\text{expected}})$ and $S(P_{\text{expected}})$ for small systems by series, and compare them with the answer given by Eq.(32) and Eq.(34)

in the limit of $N = 2^n \to \infty$. If the error is large, then it means we cannot use the simplified form Eq.(9) of $\mathcal{F}_{\text{XEB}}$ in small systems.

Set $P_{\text{incoherent}} = P_{\text{uni}}$, the uniform distribution, and $P_{\text{expected}} = P_{\text{dis}}$, the discrete Porter-Thomas distribution Eq.(39). Then we have

$$\begin{aligned} S(P_{\text{uni}}, P_{\text{dis}}) &= -\sum_{i=1}^{N} \frac{1}{N} \cdot \log \left\{ \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \right\} \\ &= \log N \cdot \left(\sum_{i=1}^{N} \frac{1}{N} \right) - \sum_{i=1}^{N} \frac{1}{N} \cdot \log \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \\ &= \log N - \sum_{i=1}^{N} \frac{1}{N} \cdot \log \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \end{aligned} \tag{45}$$

$$\begin{aligned} S(P_{\text{dis}}) &= -\sum_{x=1}^{N} \frac{\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1}}{N} \log \frac{\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1}}{N} \\ &= \log N \cdot \left\{ \sum_{x=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \right\} - \\ &\quad \sum_{x=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \cdot \log \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \\ &= \log N - \sum_{x=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \cdot \log \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \end{aligned} \tag{46}$$

To demonstrate the reliability of Eq.(32) and Eq.(34) in small N limit, compared with Eq.(45) and Eq.(46), we only need to find the difference between $-\gamma$ and

$$F_1 = \sum_{i=1}^{N} \frac{1}{N} \cdot \log \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \tag{47}$$

as well as $1-\gamma$ and

$$F_2 = \sum_{x=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \cdot \log \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right] \tag{48}$$

Table 1 shows the results of F_1 and F_2 for different $N = 2^n, n \in [1, 20]$. $S(P_{\text{uni}}, P_{\text{dis}}) = \log N - F_1$, $S(P_{\text{dis}}) = \log N - F_2$, so according to Eq.(7)

$$\mathcal{F}_{\text{XEB}} = \frac{\log N - F_1 - \langle -\log P_i \rangle_j}{\log N - F_1 - (\log N - F_2)} = \frac{\log N - F_1 - \langle -\log P_i \rangle_j}{F_2 - F_1} \tag{49}$$

If $F_1 \approx -\gamma$, $F_2 - F_1 \approx 1$, then Eq.(9) can be a good approximation with acceptible error. The closer F_1 and $F_2 - F_1$ are to $-\gamma$ and 1, the more accurate Eq.(9) can be. From the table we can see that, when $n \geq 10$, Eq.(9) can be an excellent approximation, with errors expected to be smaller than 10^{-3}, while for $5 < n < 10$, approximation is still acceptible, with errors around 10^{-2}. Only for $n \leq 3$, the true solutions deviate extremely from the large system limits, with errors scaling 10^{-1}, thus the usage of the original Eq.(7) is required. However, for random circuit sampling, we can hardly use so few qubits, so generally, Eq.(9) is a good approximation and simplification, by using which we can avoid a large amount of calculation of $S(P_{\text{expected}})$ and $S(P_{\text{incoherent}}, P_{\text{expected}})$.

n	F_1	F_2	$F_2 - F_1$	n	F_1	F_2	$F_2 - F_1$
1	-0.32739901	0.26454039	0.59193941	11	-0.57704292	0.42275459	0.99979751
2	-0.46478230	0.36250549	0.82728779	12	-0.57712956	0.42277058	0.99990014
3	-0.52517029	0.39927303	0.92444332	13	-0.57717272	0.42277793	0.99995065
4	-0.55256975	0.41335018	0.96591994	14	-0.57719423	0.42278134	0.99997557
5	-0.56534242	0.41888712	0.98422954	15	-0.57720497	0.42278293	0.99998789
6	-0.57142594	0.42113073	0.99255667	16	-0.57721032	0.42278367	0.99999399
7	-0.57436926	0.42206642	0.99643568	17	-0.57721300	0.42278402	0.99999702
8	-0.57580875	0.42246673	0.99827548	18	-0.57721433	0.42278419	0.99999852
9	-0.57651781	0.42264170	0.99915951	19	-0.57721500	0.42278426	0.99999926
10	-0.57686872	0.42271950	0.99958823	20	-0.57721533	0.42278430	0.99999963

Table 1: *As* $N = 2^n \to +\infty$, $F_1 = \sum_{i=1}^{N} \frac{1}{N} \cdot \log\left[\log\frac{N}{i} + 1 - (i-1)\log\frac{i}{i-1}\right] \to -\gamma$, $F_2 = \sum_{x=1}^{N} \frac{1}{N}\left[\log\frac{N}{i} + 1 - (i-1)\log\frac{i}{i-1}\right] \cdot \log\left[\log\frac{N}{i} + 1 - (i-1)\log\frac{i}{i-1}\right] \to 1 - \gamma$. *If we assume* $P_{\text{incoherent}} = P_{\text{uni}}$ *(uniform), and* $P_{\text{expected}} = P_{\text{dis}}$ *(discrete Porter-Thomas distribution), are ideal, we have* $S(P_{\text{uni}}, P_{\text{dis}}) = \log N - F_1$, $S(P_{\text{dis}}) = \log N - F_2$. *So* $\mathcal{F}_{\text{XEB}} = \frac{S(P_{\text{incoherent}}, P_{\text{expected}}) - S(P_{\text{measured}}, P_{\text{expected}})}{S(P_{\text{incoherent}}, P_{\text{expected}}) - S(P_{\text{expected}})} = \frac{\log N - F_1 - \langle -\log P_i\rangle_j}{\log N - F_1 - (\log N - F_2)} = \frac{\log N - F_1 - \langle -\log P_i\rangle_j}{F_2 - F_1}$.

When F_1 *and* F_2 *are very close to large* $N = 2^n$ *limits,* $-\gamma$ *and* $1 - \gamma$, *then we can simplify the* $\mathcal{F}_{\text{XEB}}$ *as Eq.*(9), *and thus avoid a large amount of calculation of* $S(P_{\text{expected}})$ *and* $S(P_{\text{incoherent}}, P_{\text{expected}})$ *to get a good approximation of* $\mathcal{F}_{\text{XEB}}$. *Euler–Mascheroni constant* $\gamma = 0.5772156649$, *so from the table we can see that, when* $n \geq 10$, *Eq.*(9) *can be an excellent approximation, with error expected to be smaller than* 10^{-3}, *while for* $5 < n < 10$, *approximation is still acceptible, with error around* 10^{-2}. *Only for* $n \leq 3$, *the true solutions deviate extremely from the large system limits, thus requiring to use the original Eq.*(7). *However, for random circuit sampling, we can hardly use so few qubits, so generally, Eq.*(9) *is a good approximation and simplification.*

A.3 Relation of Linear (Cross-)Entropy to Error Rate

$-\log p$ and $1-p$ both monotallically increase as p decrease from 1 to 0. Therefore, if we sort $p(x_i)$ in the decreasing order, $-\log p(x_i)$ and $1-p(x_i)$ are both in the increasing order. Similar to Appendix A.1, by considering the definition of two entropies Eq.(2) and Eq.(11) as the weighted sum of $-\log p(x_i)$ and $1-p(x_i)$, it is easy to understand that they will get larger as $p(x_i)$ gets flattened towards uniform distribution. With the weights $p(x_i)$ of larger values of $-\log p(x_i)$ and $1-p(x_i)$ (corresponding to smaller $p(x_i)$) getting larger, weights of smaller values getting smaller, and the total weight is $\sum_{x_i} p(x_i) = 1$ is constant, it is obvious that the weighted sum keeps increasing, until the weight $p(x_i)$ is uniform. For (linear) cross-entropy of two resemble distributions, the result is likewise, according to Appendix A.1.

Here is a simple proof that $S_{\mathrm{L}}(p)$ is the largest when distribution is uniform.

$$S_{\mathrm{L}}(p_i) = \sum_{i=1}^{N} p_i(1-p_i) = \sum_{i=1}^{N} p_i - \sum_{i=1}^{N} p_i^2 = 1 - \sum_{i=1}^{N} p_i^2 \tag{50}$$

According to the inequality of the arithmatic and geometric means

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n} a_i^2} \geq \frac{1}{n}\sum_{i=1}^{n} a_i \tag{51}$$

and the equality holds if and only if $a_i = \frac{1}{n}$, $i = 1...n$. Set $a_i = p_i$, $n = N$, then

$$\sum_{i=1}^{N} p_i^2 \geq n\left(\frac{1}{N}\sum_{i=1}^{N} a_i\right)^2 = \frac{1}{N} \tag{52}$$

Therefore,

$$S_{\mathrm{L}}(p_i) = 1 - \sum_{i=1}^{N} p_i^2 \leq 1 - \frac{1}{N} \tag{53}$$

and the equality holds if and only if $p_i = \frac{1}{N}$, $i = 1...N$, which is the uniform distribution. In other words, $S_{\mathrm{L}}(p)$ is the largest when distribution is uniform.

A.4 Linear (Cross-)Entropy of Porter-Thomas Distribution

Similar to Appendix A.2, in this part, we will calculate $S_{\mathrm{L}}(P_{\mathrm{incoherent}}, P_{\mathrm{expected}})$ and $S(P_{\mathrm{expected}})$, in the limit of $N = 2^n \to \infty$ with integration, and for small N using series. By comparing the finite system values and the large $N = 2^n$ limit, we will see in what condition the simplified Eq.(17) can be a good approximation for $\mathcal{F}_{\mathrm{LXEB}}$.

In ideal case, set $P_{\mathrm{incoherent}} = P_{\mathrm{uni}}$, the uniform distribution, and $P_{\mathrm{expected}} = P_{\mathrm{PTD}}$, the probabilities satiesfying Porter-Thomas distribution Eq.(39). Following what we did in Appendix A.2, as $N = 2^n \to +\infty$

$$
\begin{aligned}
S_{\mathrm{L}}(P_{\mathrm{uni}}, P_{\mathrm{PTD}}) &= \sum_{p_j} \left\{ \sum_{P(x_i)\in p_j} \frac{1}{N} \cdot [1 - P_{\mathrm{PTD}}(x_i)] \right\} \\
&= \sum_{p_j} \left[N \mathrm{e}^{-Np_j} \Delta p \cdot (1 - p_j) \right] \\
&\xrightarrow{\Delta p \to 0} \int_0^1 N(1-p)\mathrm{e}^{-Np} \mathrm{d}p \\
&= \int_0^1 \mathrm{e}^{-Np} \mathrm{d}(Np) - \frac{1}{N} \int_0^1 Np \cdot \mathrm{e}^{-Np} \mathrm{d}(Np) \\
&= \int_0^N \mathrm{e}^{-x} \mathrm{d}(x) - \frac{1}{N} \int_0^N x \cdot \mathrm{e}^{-x} \mathrm{d}(x) \\
&\xrightarrow{N \to \infty} \int_0^{+\infty} \mathrm{e}^{-x} \mathrm{d}(x) - \frac{1}{N} \int_0^{\infty} x \cdot \mathrm{e}^{-x} \mathrm{d}(x) \\
&= 1 - \frac{1}{N}
\end{aligned}
\tag{54}
$$

$$
\begin{aligned}
S_{\mathrm{L}}(P_{\mathrm{PTD}}) &= \sum_{p_j} \left\{ \sum_{P(x_i)\in p_j} P_{\mathrm{PTD}}(x_i) \cdot [1 - P_{\mathrm{PTD}}(x_i)] \right\} \\
&= \sum_{p_j} \left[N^2 \mathrm{e}^{-Np_j} \Delta p \cdot p_j \left(1 - p_j\right) \right] \\
&\xlongequal{\Delta p \to 0} \int_0^1 N^2 p(1-p) \mathrm{e}^{-Np} \mathrm{d}p \\
&= \int_0^1 Np \cdot \mathrm{e}^{-Np} \mathrm{d}(Np) - \frac{1}{N} \int_0^1 (Np)^2 \cdot \mathrm{e}^{-Np} \mathrm{d}(Np) \\
&= \int_0^N x \mathrm{e}^{-x} \mathrm{d}(x) - \frac{1}{N} \int_0^N x^2 \cdot \mathrm{e}^{-x} \mathrm{d}(x) \\
&\xrightarrow{N \to \infty} \int_0^{+\infty} x \mathrm{e}^{-x} \mathrm{d}(x) - \frac{1}{N} \int_0^{\infty} x^2 \cdot \mathrm{e}^{-x} \mathrm{d}(x) \\
&= 1 - \frac{2}{N}
\end{aligned}
\tag{55}
$$

Plug Eq.(54) and Eq.(55) into Eq.(15), then we will get the simplified formula Eq.(17).

For small $N = 2^n$, the linear (cross-)entropy should be writen in series

$$
S_{\mathrm{L}}(P_{\mathrm{uni}}, P_{\mathrm{dis}}) = \langle 1 - P_{\mathrm{dis}}(x_i) \rangle_{\mathrm{uni},j} = \sum_{i=1}^{N} \frac{1}{N} \cdot [1 - P_{\mathrm{dis}}(x_i)] = 1 - \frac{1}{N} \left[\sum_{i=1}^{N} P_{\mathrm{dis}}(x_i) \right] = 1 - \frac{1}{N}
\tag{56}
$$

$$
\begin{aligned}
S_{\mathrm{L}}(P_{\mathrm{dis}}) =& \langle 1 - P_{\mathrm{dis}}(x_i) \rangle_{\mathrm{dis},j} = \sum_{i=1}^{N} P_{\mathrm{dis}}(x_i) \cdot [1 - P_{\mathrm{dis}}(x_i)] = 1 - \sum_{i=1}^{N} \left[P_{\mathrm{dis}}(x_i) \right]^2 \\
=& 1 - \frac{1}{N} \left\{ \sum_{i=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right]^2 \right\}
\end{aligned}
\tag{57}
$$

Compared with Eq.(55), there should be

$$
\lim_{N \to +\infty} \sum_{i=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right]^2 = 2
\tag{58}
$$

Since $S_{\mathrm{L}}(P_{\mathrm{uniform}}, P_{\mathrm{expected}}) = 1 - \frac{1}{N}$ is always true, even when P_{expected} is an arbitrary distribution, to demonstrate the reliability of Eq.(55) in small $N = 2^n$ limit, and thus the accuracy of Eq.(17), we only need to find the difference between 2 and

$$F_3 = \sum_{i=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right]^2 \tag{59}$$

Then $S_{\mathrm{L}}(P_{\mathrm{dis}}) = 1 - \frac{F_3}{N}$, and a more accurate formula for $\mathcal{F}_{\mathrm{LXEB}}$ is

$$\begin{aligned} \mathcal{F}_{\mathrm{LXEB}} =& \frac{1 - \frac{1}{N} - (1 - \langle P_{\mathrm{expected}}(x_i) \rangle_{\mathrm{measured},j})}{1 - \frac{1}{N} - (1 - \frac{F_3}{N})} \\ =& \frac{N \langle P_{\mathrm{expected}}(x_i) \rangle_{\mathrm{measured},j} - 1}{F_3 - 1} \end{aligned} \tag{60}$$

In theory, this formula, ought to work for any size of systems, is more reliable than Eq.(17). While in real experiment, if we do not want to assume beforehead that P_{expected} and the discrete Porter-Thomas distribution Eq.(39) match exactly, we will need to calculate this $F_3 = N \langle P_{\mathrm{expected}}(x_i) \rangle_{\mathrm{expected},j}$ from simulated outputs. Then the exact formula for $\mathcal{F}_{\mathrm{LXEB}}$ all by experiment, simulation and real quantum processing, for any size of system, (noticing $P_{\mathrm{incoherent}} = P_{\mathrm{uniform}}$,) is

$$\begin{aligned} \mathcal{F}_{\mathrm{LXEB}} =& \frac{1 - \frac{1}{N} - (1 - \langle P_{\mathrm{expected}}(x_i) \rangle_{\mathrm{measured},j})}{1 - \frac{1}{N} - (1 - \frac{F_3}{N})} \\ =& \frac{N \langle P_{\mathrm{expected}}(x_i) \rangle_{\mathrm{measured},j} - 1}{N \langle P_{\mathrm{expected}}(x_i) \rangle_{\mathrm{expected},j} - 1} \end{aligned} \tag{61}$$

Table 2 shows the results of F_3 for different $N = 2^n, n = [1, 20]$. If $F \approx 2$, then according to Eq.(60), Eq.(17) can be a good approximation for $\mathcal{F}_{\mathrm{LXEB}}$ with acceptible error. The closer F_3 is to 2, the more accurate Eq.(17) can be. From

the table we can see that, when $n \geq 10$, Eq.(17) can be an excellent approximation, with error expected to be smaller than 10^{-3}, while for $5 < n < 10$, approximation is still acceptible, with error around 10^{-2}. Only for $n \leq 3$, the true solutions deviate extremely from the large system limits, when errors are at the scale 10^{-1}, thus requiring to use the original Eq.(15), which in a clearer form is Eq.(61).

If we choose to assume that P_{expected} follows Porter-Thomas distribution, we can also replace $N\langle P_{\text{expected}}(x_i)\rangle_{\text{expected},j}$ with F_3 from Table 2, which is Eq.(60). However, for random circuit sampling, we can hardly use so few qubits, so generally, Eq.(17) is a good approximation and simplification, by using which we can avoid a large amount of calculation of $S_{\text{L}}(P_{\text{expected}})$. And $S_{\text{L}}(P_{\text{incoherent}}, P_{\text{expected}}) = 1 - \frac{1}{N}$ is always true, while $P_{\text{incoherent}} = P_{\text{uniform}}$, so unlike cross-entropy benchmarking, we don't need to calculate this linear cross-entropy.

Table 2 is calculated from a simple program. When running this program, I found an interesting phenomena about the series calculating

$$\begin{aligned} L_{N,\text{dis}}(k) &= \sum_{i=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right]^k \\ &= \sum_{i=1}^{N} \frac{1}{N} \left[\log N + 1 - i \log i + (i-1) \log(i-1) \right]^k \end{aligned} \tag{62}$$

n	F_3	n	F_3	n	F_3	n	F_3
1	1.48045301	6	1.98314023	11	1.99947252	16	1.99998352
2	1.73511236	7	1.99156503	12	1.99973625	17	1.99999176
3	1.86626030	8	1.99578124	13	1.99986813	18	1.99999588
4	1.93280502	9	1.99789030	14	1.99993406	19	1.99999794
5	1.96632116	10	1.99894507	15	1.99996703	20	1.99999897

Table 2: $\lim\limits_{N\to+\infty} F_3 = \lim\limits_{N\to+\infty} \sum\limits_{i=1}^{N} \frac{1}{N}\left[\log\frac{N}{i} + 1 - (i-1)\log\frac{i}{i-1}\right]^2 = 2$, $N = 2^n$. *If we assume* $P_{\text{incoherent}} = P_{\text{uni}}$ *(uniform), and* $P_{\text{expected}} = P_{\text{dis}}$ *(discrete Porter-Thomas distribution), are ideal, we have* $S_{\text{L}}(P_{\text{uni}}, P_{\text{dis}}) = 1 - \frac{1}{N}$, $S_{\text{L}}(P_{\text{dis}}) = 1 - \frac{F_3}{N}$. *So* $\mathcal{F}_{\text{LXEB}} = \frac{S_{\text{L}}(P_{\text{incoherent}},P_{\text{expected}})-S_{\text{L}}(P_{\text{measured}},P_{\text{expected}})}{S_{\text{L}}(P_{\text{incoherent}},P_{\text{expected}})-S_{\text{L}}(P_{\text{expected}})} = \frac{1-\frac{1}{N}-(1-\langle P_i\rangle_j)}{1-\frac{1}{N}-(1-\frac{F_3}{N})} = \frac{N\langle P_i\rangle_j-1}{F_3-1}$.

When $F_3 \to 2$, *we can simplify the* $\mathcal{F}_{\text{LXEB}}$ *as Eq.*(17), *and thus avoid a large amount of calculation of* $S_{\text{L}}(P_{\text{expected}})$ *but still get a good approximation of* $\mathcal{F}_{\text{LXEB}}$. *So from the table we can see that, when* $n > 10$, *Eq.*(17) *can be an excellent approximation, with error expected to be smaller than* 10^{-3}, *while for* $5 < n \leq 10$, *approximation is still acceptible, with error around* 10^{-2}. *Only for* $n \leq 3$, *the true solutions deviate extremely from the large system limits, with error over* 10^{-1}, *thus requiring to use the original Eq.*(15). *However, for random circuit sampling, we can hardly use so few qubits, so generally, Eq.*(17) *is a good approximation and simplification.*

For $k = 1$, it is just the total probability of P_{dis} (Eq.(39)), so $L_{N,\text{dis}}(1) = 1$. And for $L_{N,\text{dis}}(2) = F_3$, we have found from Table 2 that $\lim_{N\to+\infty} L_{N,\text{dis}}(2) = 2$. It is interesting to see that by changing the power k within integer values, such series can provide another integer limit. Therefore, more values of k are tried for a maximum of $N = 2^{20}$. For $k = 2, ..., 6$, the outputs are $L_{N,\text{dis}}(k) = 1.99999897$, 5.99995263, 23.99853818, 119.96215938, 719.11235603, which are approximately 2, 6, 24, 120, 720, or say 2!, 3!, 4!, 5!, and 6!. For $k = 7$, $L_{N,\text{dis}}(k) = 5020.42545926$ is also close to $7! = 5040$. So it can be easily guessed that for integer k

$$\lim_{N\to+\infty} L_{N,\text{dis}}(k) = k! \tag{63}$$

Our most familiar function who can output factorial is $\Gamma(x)$. For integer n, $\Gamma(n) = (n-1)!$. Therefore, a conjecture is that for all real k's

$$\lim_{N\to+\infty} L_{N,\text{dis}}(k) = \Gamma(k+1) \tag{64}$$

To briefly indicate that this conjecture may be true, more values of k are checked, from 0 to 2, with interval width 0.1. The result matches perfectly with $\Gamma(k+1)$, as Table 3 shows.

For LXEB, since there is not a $\log p$ term, or such terms that converge to infinity as $p \to 0$, in the series of linear (cross-)entropy, therefore, as we have proved that for large $N = 2^n$, $P_{\text{PTD,discrete}} \to P_{\text{PTD}}$, we can directly use $P_{\text{PTD}}(x) = \frac{1}{N}\log\frac{N}{x}$ as the probabilities for discrete i's in the limit of $N = 2^n \to +\infty$ in LXEB, without the concern of convergence by the $p = 0$ term. From Eq.(57) and Eq.(59), we know

$$F_3 = N\langle P_{\text{expected}}(x_i)\rangle_{\text{expected},j} = \sum_{i=1}^{N} N \cdot [P_{\text{expected}}(x_i)]^2 \tag{65}$$

k	$L_{N,\mathrm{dis}}(k)$	$\Gamma(k+1)$	k	$L_{N,\mathrm{dis}}(k)$	$\Gamma(k+1)$
-0.1	1.06862854	1.06862870	1.0	1.00000000	1.00000000
0.0	1.00000000	1.00000000	1.1	1.04648584	1.04648585
0.1	0.95135078	0.95135077	1.2	1.10180248	1.10180249
0.2	0.91816875	0.91816874	1.3	1.16671188	1.16671191
0.3	0.89747070	0.89747070	1.4	1.24216929	1.24216934
0.4	0.88726382	0.88726382	1.5	1.32934029	1.32934039
0.5	0.88622693	0.88622693	1.6	1.42962439	1.42962456
0.6	0.89351535	0.89351535	1.7	1.54468558	1.54468585
0.7	0.90863874	0.90863873	1.8	1.67649036	1.67649079
0.8	0.93138377	0.93138377	1.9	1.82735441	1.82735508
0.9	0.96176583	0.96176583	2.0	1.99999897	2.00000000

Table 3: *The value of series* $L_{N,\mathrm{dis}}(k) = \sum_{i=1}^{N} \frac{1}{N} \left[\log \frac{N}{i} + 1 - (i-1) \log \frac{i}{i-1} \right]^k$, *with* $N = 2^{20}$, *compared with* $\Gamma(k)$. *From this table and the other results for* $k = 3, 4, 5, 6, 7$, *We can conject that* $\lim_{N \to +\infty} L_{N,\mathrm{dis}}(k) = \Gamma(k)$. *Strict proof is attached in the following.*

Use P_{PTD} as P_{expected}, then it can be expected that as $N = 2^n \to +\infty$, the series will still converge to 2.

$$F_3(P_{\mathrm{PTD}}) = \sum_{i=1}^{N} N \cdot [P_{\mathrm{PTD}}(x_i)]^2 = \sum_{i=1}^{N} \frac{1}{N} \cdot \left[\log \left(\frac{N}{i} \right) \right]^2 \tag{66}$$

Without the correction term Δp_i on P_{PTD}, Eq.(66) looks much simpler than Eq.(59). But as we have proved, as $N = 2^n \to \infty$, for almost all i's, Δp_i can be neglected. And for the only several exceptions with i very close to N, the corresponding

terms in Eq.(66) are

$$\frac{1}{N}\left[\log\left(\frac{N}{i}\right)\right]^2 \approx \frac{1}{N}\left(\frac{N}{i}-1\right)^2 \approx \frac{(N-i)^2}{N^3} \tag{67}$$

Since the number of these $a = N - i \ll N$, the sum of them

$$\sum_{N-i=1}^{a} \frac{(N-i)^2}{N^3} \sim \left(\frac{a}{N}\right)^3 \tag{68}$$

As discussed, for all N, $a \sim 1$, so the sum of such terms in P_{PTD}

$$\sum_{N-i=1}^{a} \frac{(N-i)^2}{N^3} \sim N^{-3} \tag{69}$$

The corresponding terms in Eq.(59) with non-negligible Δp_i following Eq.(43)

$$\frac{1}{N}\left[\log\left(\frac{N}{i}\right)+N\Delta p_i\right]^2 \approx \frac{1}{N}\left(\frac{N}{i}-1+\frac{1}{2i}\right)^2 \approx \frac{(N-i+\frac{1}{2})^2}{N^3} \tag{70}$$

$$\sum_{N-i=1}^{a} \frac{(N-i+\frac{1}{2})^2}{N^3} \sim \left(\frac{a+\frac{1}{2}}{N}\right)^3 \sim N^{-3} \tag{71}$$

Therefore, the difference of the sum of the terms with non-negligible Δp_i in Eq.(59) and Eq.(66) should be no more than the scale of N^{-3}, which approaches 0 in the limit $N \to +\infty$, and thus can also be neglected. To conclude, in large system limit, the difference of F_3 computed from Eq.(59) and Eq.(66) should be negligible, which means they should both converge to the same value 2.

Furthermore, since P_{PTD} and P_{dis} act identically in large N limit, the new series from continuous Porter-Thomas distribution Eq.(37) should also converge to $\Gamma(k+1)$

$$\lim_{N\to+\infty} L_{N,\mathrm{con}}(k) = \lim_{N\to+\infty} \sum_{i=1}^{N} \frac{1}{N}\left(\log\frac{N}{i}\right)^k = \Gamma(k+1) \tag{72}$$

$L_{N,\mathrm{con}}(k)$ is also calculated in the same way as $L_{N,\mathrm{dis}}(k)$ and the result shows that our conjecture is correct. To avoid duplication, the table of $L_{N,\mathrm{con}}(k)$ is skipped.

Compared with $L_{N,\mathrm{dis}}(k)$, it is easier to prove that $L_{N,\mathrm{con}}(k)$ converges to $\Gamma(k+1)$, then since $P_{\mathrm{dis}} \xrightarrow{N\to\infty} P_{\mathrm{PTD}}$, so they will converge to the same function, i.e. $\Gamma(k+1)$.

When $N \to +\infty$, the sum in Eq.(72) can be writen as integral

$$\lim_{N\to+\infty} L_{N,\mathrm{con}}(k) = \int_0^N \frac{1}{N}\left(\log\frac{N}{x}\right)^k \mathrm{d}x \tag{73}$$

Change the variable to $y = \log\frac{N}{x}$, $\mathrm{d}x = -N\mathrm{e}^{-y}\mathrm{d}y$, then

$$\lim_{N\to+\infty} L_{N,\mathrm{con}}(k) = \int_0^{+\infty} y^k \mathrm{e}^{-y}\mathrm{d}y \tag{74}$$

The definition of Gamma function is

$$\Gamma(z) = \int_0^{+\infty} x^{z-1}\mathrm{e}^{-x}\mathrm{d}x \tag{75}$$

So apparently

$$\lim_{N\to+\infty} L_{N,\mathrm{con}}(k) = \Gamma(k+1) \tag{76}$$

and so is $L_{N,\mathrm{dis}}(k)$.

We notice that if we define new k-(cross-)entropy as

$$S_k(p(x_i)) = \sum_{i=1}^{N} p(x_i)\cdot\left\{1-[p(x_i)]^k\right\} \tag{77}$$

$$S_k(p(x_i), q(x_i)) = \sum_{i=1}^{N} p(x_i)\cdot\left\{1-[q(x_i)]^k\right\} \tag{78}$$

Then likewise, for small systems,

$$S_k(P_{\mathrm{uni}}, P_{\mathrm{dis}}) = \langle 1-[P_{\mathrm{dis}}(x_i)]^k\rangle_{\mathrm{uni},j} = \sum_{i=1}^{N}\frac{1}{N}\cdot\left\{1-[P_{\mathrm{dis}}(x_i)]^k\right\} = 1-\frac{1}{N}\sum_{i=1}^{N}[P_{\mathrm{dis}}(x_i)]^k \tag{79}$$

$$S_k(P_{\rm dis}) = \langle 1 - [P_{\rm dis}(x_i)]^k \rangle_{{\rm dis},j} = \sum_{i=1}^{N} P_{\rm dis}(x_i) \cdot \left\{ 1 - [P_{\rm dis}(x_i)]^k \right\} = 1 - \sum_{i=1}^{N} [P_{\rm dis}(x_i)]^{k+1} \tag{80}$$

And for large systems

$$\begin{aligned} S_k(P_{\rm uni}, P_{\rm PTD}) &= \sum_{p_j} \left\{ \sum_{P(x_i) \in p_j} \frac{1}{N} \cdot \left\{ 1 - [P_{\rm PTD}(x_i)]^k \right\} \right\} = 1 - \sum_{p_j} N {\rm e}^{-Np_j} \Delta p \cdot p_j^k \\ &\xrightarrow{\Delta p \to 0} 1 - \int_0^1 N p^k {\rm e}^{-Np} {\rm d}p \overset{x=Np}{=\!=} 1 - \frac{1}{N^k} \int_0^N x^k {\rm e}^{-x} {\rm d}(x) \\ &\xrightarrow{N \to \infty} 1 - \frac{1}{N^k} \int_0^{+\infty} x^k {\rm e}^{-x} {\rm d}(x) = 1 - \frac{\Gamma(k+1)}{N^k} \end{aligned} \tag{81}$$

$$\begin{aligned} S_k(P_{\rm PTD}) &= \sum_{p_j} \left\{ \sum_{P(x_i) \in p_j} P_{\rm PTD}(x_i) \cdot \left\{ 1 - [P_{\rm PTD}(x_i)]^k \right\} \right\} = 1 - \sum_{p_j} N^2 {\rm e}^{-Np_j} \Delta p \cdot p_j^{k+1} \\ &\xrightarrow{\Delta p \to 0} 1 - \int_0^1 N^2 p^{k+1} {\rm e}^{-Np} {\rm d}p \overset{x=Np}{=\!=} 1 - \frac{1}{N^k} \int_0^N x^{k+1} {\rm e}^{-x} {\rm d}(x) \\ &\xrightarrow{N \to \infty} 1 - \frac{1}{N^k} \int_0^{+\infty} x^{k+1} {\rm e}^{-x} {\rm d}(x) = 1 - \frac{\Gamma(k+2)}{N^k} \end{aligned} \tag{82}$$

So if we define k'th order cross-entropy benchmarking fidelity as the following

$$\mathcal{F}_{\rm kXEB} = \frac{S_k(P_{\rm incoherent}, P_{\rm expected}) - S_k(P_{\rm measured}, P_{\rm expected})}{S_k(P_{\rm incoherent}, P_{\rm expected}) - S_k(P_{\rm expected})} \tag{83}$$

since

$$S_k(p(x_i), q(x_i)) = \sum_{i=1}^{N} p(x_i) \cdot \left[1 - q(x_i)^k \right] = 1 - \langle q(x_i)^k \rangle_p \tag{84}$$

then Eq.(83) in real experiment is calculated by

$$\mathcal{F}_{\rm kXEB} = \frac{\langle [P_{\rm expected}(x_i)]^k \rangle_{\rm measured} - \langle [P_{\rm expected}(x_i)]^k \rangle_{\rm incoherent}}{\langle [P_{\rm expected}(x_i)]^k \rangle_{\rm expected} - \langle [P_{\rm expected}(x_i)]^k \rangle_{\rm incoherent}} \tag{85}$$

For large systems, assume $P_{\text{expected}} = P_{\text{PTD}}$, $P_{\text{incoherent}} = P_{\text{uniform}}$, and from Eq.(81) and Eq.(82), we know that

$$\begin{aligned} \lim_{N\to+\infty} \langle P_{\text{PTD}}^k \rangle_{\text{uni}} &= \frac{\Gamma(k+1)}{N^k} \\ \lim_{N\to+\infty} \langle P_{\text{PTD}}^k \rangle_{\text{PTD}} &= \frac{\Gamma(k+2)}{N^k} \end{aligned} \tag{86}$$

so in such limit, Eq.(83) can be simplified as

$$\begin{aligned} \mathcal{F}_{\text{kXEB}} &= \frac{N^k \langle [P_{\text{expected}}(x_i)]^k \rangle_{\text{measured}} - \Gamma(k+1)}{\Gamma(k+2) - \Gamma(k+1)} \\ &= \frac{N^k \langle [P_{\text{expected}}(x_i)]^k \rangle_{\text{measured}} - \Gamma(k+1)}{(k+1)\Gamma(k+1) - \Gamma(k+1)} \\ &= \frac{1}{k} \left[\frac{N}{\Gamma(k+1)} \langle [P_{\text{expected}}(x_i)]^k \rangle_{\text{measured}} - 1 \right] \end{aligned} \tag{87}$$

We notice that for $k = 1$, i.e. LXEB, Eq.(87) is just the $\mathcal{F}_{\text{LXEB}}$ we used Eq.(17).

For small systems, we can still do as discussed, use the original definition Eq.(83), or calculate the series Eq.(81) and Eq.(82) of ideal distribution P_{dis} for small systems, to modify Eq.(87) by Eq.(83).

As a generalization of Appendix A.1 and A.3, $1 - p^k, (k > 0)$ and $-\log p$ have similar properties, and the k'th order (cross-)entropy can be qualified as a quantification of the degree of disorder and error rate. Here k does not have to be an integer. Or actually, any function $s(p)$ defined in the domain $p \in [0, 1]$, which monotonically decreases to 0 at $p = 1$, can be used to define somewhat an entropy $S_s(p) = \sum_{i=1}^{N} p \cdot s(p)$, and likewise, corresponding cross-entropy $S_s(p, q)$ and fidelity $\mathcal{F}_{\text{sXEB}}$, although the behavior of such fidelity with respect to error rate may vary, according to specific function $s(p)$.

References

[1] Shor, P. W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* **26**(5), 1484-1509 (1997) https://doi.org/10.1137/S0097539795293172.

[2] Chris Lomot. The Hidden Subgroup Problem - Review and Open Problems. Preprint at https://arxiv.org/abs/quant-ph/0411037 (2004).

[3] Georgescu, I. M., Ashhab, S., & Nori, Franco. Quantum simulation. *Rev. Mod. Phys.* **86**(1), 153-185 (2014) https://doi.org/10.1103/RevModPhys.86.153.

[4] Lund, A.P., Bremner, M. J. & Ralph, T. C. Quantum sampling problems, BosonSampling and quantum supremacy. *npj Quantum Inf* **3**, 15 (2017) https://doi.org/10.1038/s41534-017-0018-2.

[5] Aaronson, S., & Arkhipov, A. The Computational Complexity of Linear Optics. In *STOC '11: Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing* 333-342 (ACM 2011) https://doi.org/10.1145/1993636.1993682. *Theory Comput.* **9**(4), 143-252 (2013) https://doi.org/10.1145/1993636.1993682. Preprint at https://arxiv.org/abs/1011.3245 (2010).

[6] Brod, D. J. *et al.* Photonic implementation of boson sampling: a review. *Adv. Photon.* **1**(3), 034001 (2019) https://doi.org/10.1117/1.AP.1.3.034001.

[7] Arute, F. *et al.* Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505-510 (2019) https://doi.org/10.1038/s41586-019-1666-5.

[8] https://qiskit.org/.

[9] Gubin, A. & Santos, L. F. Quantum chaos: An introduction via chains of interacting spins 1/2. *Am. J. Phys.* **80**(3), 246-251 (2012) https://doi.org/10.1119/1.3671068.

[10] Boixo, S., Isakov, S. V., & Smelyanskiy, V. N. *et al.* Characterizing quantum supremacy in near-term devices. *Nat. Phys.* **14**, 595–600 (2018) https://doi.org/10.1038/s41567-018-0124-x.

[11] Bouland A., Fefferman B., Nirkhe C., & Vazirani U. On the complexity and verification of quantum random circuit sampling. *Nat. Phys.* **15**, 159–163 (2019) https://doi.org/ 10.1038/s41567-018-0318-2.

[12] Neill, C. *et al.* A blueprint for demonstrating quantum supremacy with superconducting qubits. *Science* **360**(6385), 195-199 (2018) https://doi.org/10.1126/science.aao4309.

[13] https://quantum-computing.ibm.com/.