The power of random quantum circuits

Bill Fefferman

(University of Chicago)

Based on "**On the complexity and verification of quantum random circuit sampling**" with A. Bouland, C. Nirkhe, U. Vazirani (Nature Physics **15**, pages 159–163, arXiv: 1803.04402)

And "**Efficient classical simulation of noisy random quantum circuits in one dimension**" with K. Noh and L. Jiang (arXiv: 2003.13163)

Quantum advantage in the NISQ era

- We've arrived at an era in which existing quantum experiments can solve problems that *seem challenging* for classical computers
- That is, experiments are now large enough so that *best known* classical simulation techniques take large amount of time on classical supercomputers
- At the same time, these experiments have limitations, which could potentially be exploited by faster classical algorithms
  - e.g., restricted depth, uncorrected noise

Artist rendition of Google's "Sycamore" 53 qubit processor (Photo Credit: Google AI Blog)

"Quantum supremacy"

- First goal for the NISQ era: "quantum supremacy"
- "Quantum supremacy" is multifaceted – Need to find a task that simultaneously:
  1. Can be solved experimentally
  2. Is "classically hard"
     - Good theoretical (asymptotic) hardness evidence from complexity theory
     - Also cannot be solved in a "comparable" amount of time by classical supercomputer
  3. Has a procedure for verification

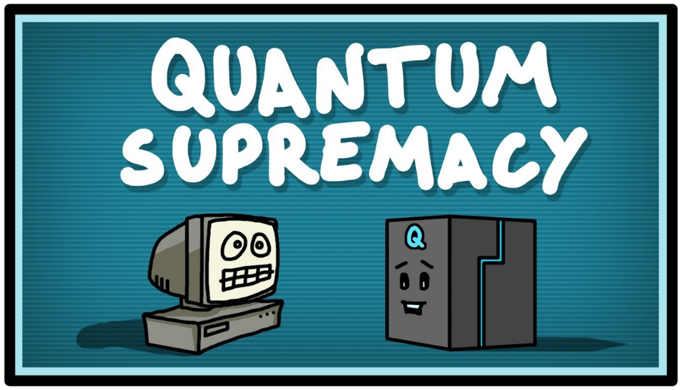
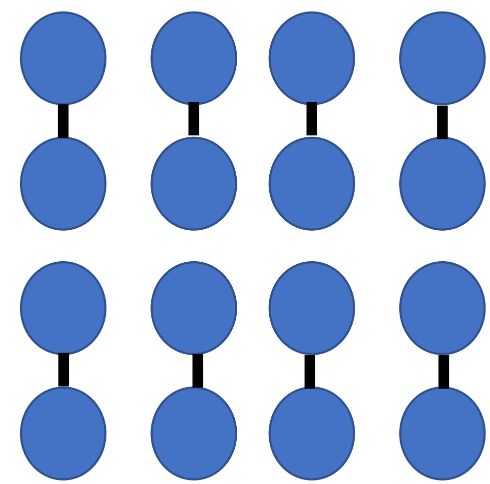


Photo Credit: "Domain of Science"

Random Quantum Circuit Sampling (RCS)

- Google's approach: *Random Circuit Sampling* [Boixo et. al. 2017, Arute et. al. 2019]
- Generate a quantum circuit C on n qubits on a 2D lattice, with $d \sim \sqrt{n}$ layers of (Haar) random nearest-neighbor gates
  - In practice use a discrete approximation to the Haar random distribution
- Start with $|0^n\rangle$ input state, apply random quantum circuit and measure in computational basis



(single layer of Haar random two qubit gates applied on 2D grid of qubits)

Why are Random Circuits an attractive proposal?

- Experimentally feasible
  - Hardness at comparatively low depth and system size
- Advantages for verification/benchmarking
  - Output distribution of random circuits have "Porter-Thomas" property
    - For any outcome x, $\Pr_C\left[|\langle x|C|0^n\rangle|^2 = \frac{q}{N}\right] \sim e^{-q}$
  - We can use this property to calculate the ideal score of a random circuit on benchmarking tests (e.g., to understand the ideal "cross-entropy" score)

Why is RCS hard classically?

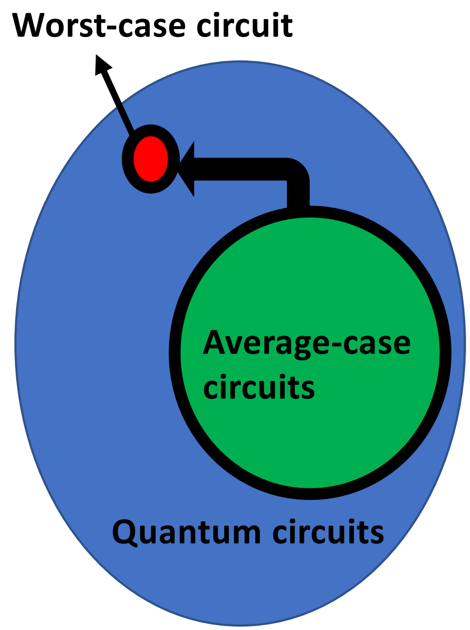
- There is good evidence that RCS is classically hard in the noiseless case (e.g., [Terhal & DiVincenzo'04][Bremner, Jozsa & Shepherd'10][Aaronson & Arkhipov '12][Aaronson & Chen'17], [BFNV'19]…)
- Some of these arguments may "carry over" to the noisy case, but we're less certain – generally requires nonstandard hardness conjectures…

Today's focus: hardness of computing output probabilities of (*noisy*) random circuits

- As compared with sampling…
  - Experiments can't efficiently solve this problem
  - Possible to prove much stronger hardness results
  - In practice, many classical simulation algorithms do compute output probabilities, so these hardness results are barriers for these algorithms
- **Agenda:**
  1. We'll review the average-case **#P**-hardness for near-exact computation of the output probability of random quantum circuits [BFNV'19]
  2. We'll show that these results still hold if the circuit is noisy (wrt a fixed noise model, e.g., local depolarizing noise) [ongoing joint work]
  3. We'll talk about new classical simulation results for 1D noisy random quantum circuits [Noh, Jiang, F'20]

Hardness of average-case problem [F, with Bouland, Nirkhe and Vazirani'19]

- **Random Quantum Circuit Output Computation:**
  - **Input**: Random quantum circuit C
  - **Output**: Compute output probability, $p_{0^n}(C) = |\langle 0^n|C|0^n\rangle|^2$ with probability $1-\delta$ over C
- To prove this is **#P**-hard we give a *worst-case to average-case reduction*
  - We build on result of [Lipton'91, AA'11] on average-case hardness of computing the **Permanent** of a matrix



Average case hardness for **Permanent** [Lipton '91]

- **Permanent** of $n \times n$ matrix is **#P**-hard in the worst-case [Valiant '79]
  - $Per[X] = \sum_{\sigma \in S_n} \prod_{i=1}^{n} X_{i,\sigma(i)}$
- *Algebraic property*: $Per[X]$ is a degree n polynomial with n^2 variables
- Need compute $Per[X]$ of worst-case matrix X
  - But we only have access to algorithm O that correctly computes *most* permanents over $\mathbb{F}_p$
  - i.e., $\Pr_{Y \in_R \mathbb{F}_p^{n \times n}}[O(Y) = Per[Y]] \geq 1 - \frac{1}{poly(n)}$
- Choose $n + 1$ fixed non-zero points $t_1, t_2 \ldots, t_{n+1} \in \mathbb{F}_p$ and uniformly random matrix R
- Consider line $A(t) = X + tR$
  - *Observation 1 "marginal property"*: for each i, $A(t_i)$ is a random matrix over $\mathbb{F}_p^{n \times n}$
  - *Observation 2*: "univariate polynomial": $Per[A(t)]$ is a degree n polynomial in t
- But now these $n + 1$ evaluation points uniquely define the polynomial, so use error-correction (i.e., polynomial interpolation) and evaluate $Per[A(0)] = Per[X]$

[BFNV'18]: Hardness for Random Quantum Circuits

- *Algebraic property*: much like $Per[X]$, output probability of random quantum circuits have low-degree polynomial structure
  - Consider circuit $C = C_m C_{m-1} \dots C_1$
  - Polynomial structure comes from Feynman path integral:
    - $\langle 0^n | C | 0^n \rangle = \sum_{y_2, y_3, \dots, y_m \in \{0,1\}^n} \langle 0^n | C_m | y_m \rangle \langle y_m | C_{m-1} | y_{m-1} \rangle \dots \langle y_2 | C_1 | 0^n \rangle$
- This is a polynomial of degree m in the gate entries of the circuit
- So the output probability $p_{0^n}(C)$ is a polynomial of degree $2m$

Worst-to-Average Reduction – Attempt 1: Copy Lipton's proof

- Our case: want to compute $p_{0^n}(C)$ for worst case C
  - But we only have the ability to compute output probabilities for *most* circuits
- *Recall*: Lipton wanted to compute $Per[X]$, choose random R, considered line $A(t) = X + tR$
- *Problem*: can't just perturb gates in a random linear direction
  - i.e., if C is unitary, D is unitary, $C + tD$ is not generally unitary

New approach to *scramble* gates of fixed circuit

- Choose and fix $\{H_i\}_{i\in[m]}$ Haar random gates
- Now consider new circuit $C' = C'_m C'_{m-1} \ldots C'_1$ so that for each gate $C'_i = C_i H_i$
  - Notice that each gate in C' is completely random – "marginal property"
- **Problem:** no univariate polynomial structure connects worst-case circuit C with the new circuit C' !!

Correlating via *quantumness*

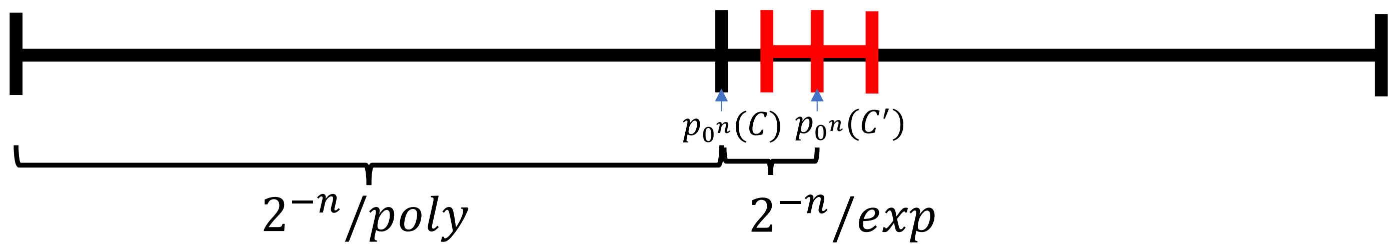
- We need the analogue to Lipton's "*univariate* polynomial structure"
- ***Main idea***: "Implement tiny fraction of H_i^{-1}"
  - i.e., $C_i' = C_i H_i e^{-ih_i\theta}$
  - If $\theta = 1$ the corresponding circuit $C' = C$, and if $\theta \approx small$, each gate is close to Haar random
  - Now take several non-zero but small θ and apply polynomial interpolation

This is still not the “right way” to scramble!

- *Problem*: $e^{-ih_i\theta}$ is not polynomial in θ
- *Solution:* take fixed truncation of Taylor series for $e^{-ih_i\theta}$
    - i.e., each gate of C' is $C_i H_i \sum_{k=0}^{K} \frac{(-ih_i\theta)^k}{k!}$
    - So each gate entry is a polynomial in θ and so is $p_{0^n}(C')$
    - Now interpolate and compute $q(1) = p_{0^n}(C)$

Understanding the [BFNV’19] construction

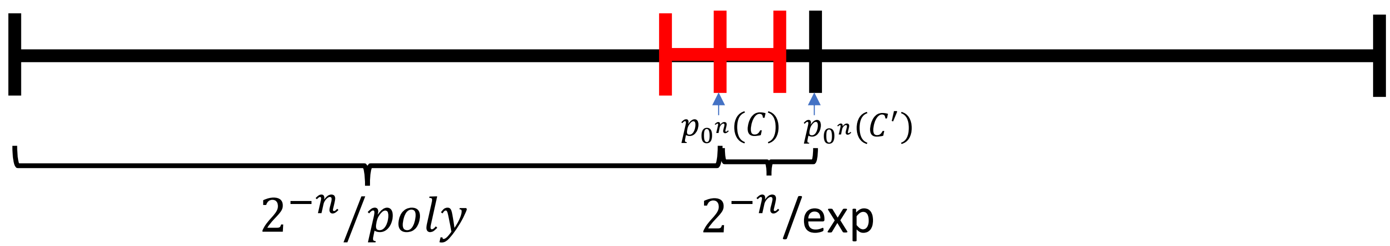
- **First point:** Polynomial interpolation is *very sensitive* to additive error
  - Severely constrains the robustness of this argument.



  - As a result we can only show hardness for any point on the red line ($p_{0^n}(C') \pm 2^{-n^3}$)
  - To prove hardness of ***sampling***, it suffices to show that computing any *point on the black interval* is **#P**-hard [Stockmeyer’83]
- **Second point:** Truncated circuit C' is *slightly* non-unitary
  - We show wrt ”hardness of sampling” level of approximation, the truncations don’t matter
  - i.e., $p_{0^n}(C) \pm \frac{2^{-n}}{poly}$ is hard to compute iff $p_{0^n}(C') \pm \frac{2^{-n}}{poly}$ is hard

Movassagh’s result

- In recent follow-up work, hardness has been shown around original output probability
  - i.e., computing $p_0(C) \pm 2^{-n^3}$ is **#P**-hard [Movassagh ’20]



- To do this, Movassagh gives a new method to interpolate between the worst-case and random quantum circuit, using the “Cayley path”, which stays unitary throughout the entire path

Is it hard to (nearly exactly) compute ***noisy*** random circuit probabilities? [*ongoing joint work*]

- **Fact**: output distribution of noisy quantum circuit converges rapidly to uniform [e.g., Aharonov, Ben-Or, Impagliazzo & Nisan '96, Gao & Duan '18…]
- **Intuition**: quantum hardness is present in tiny *deviations* from uniform
- **How to formalize?** Suppose we fix a noise model:
  - Each ideal gate C_i is followed by two qubit depolarizing noise
    - $\mathcal{E}_i = (1-q)\rho + \frac{q}{15}\sum_{\alpha,\beta \in \mathcal{P}\times\mathcal{P}-(I,I)}(\sigma_\alpha \otimes \sigma_\beta)\rho(\sigma_\alpha \otimes \sigma_\beta)$
  - That is, we can think about choosing a noisy random circuit by:
    - First pick ideal circuit $C = C_m C_{m-1} \ldots C_1$ from the random circuit distribution
    - Then environment chooses operators N, from a distribution $\mathcal{N}$ (specified by the channel)
    - We get a sample from output distribution of $N \cdot C$ without learning the noise operators

Noisy circuit output probability

- Then, by linearity, can write the output probability of the noisy circuit as:
  - $E_{N\sim\mathcal{N}}[|\langle 0^n|N\cdot C|0^n\rangle|^2] = E_{N\sim\mathcal{N}}[p_{0^n}(N\cdot C)]$
- This can be written as a weighted sum of Feynman path integrals:
  - $\sum_N \Pr_{\mathcal{N}}[N]\cdot \left|\sum_{y_1,y_2,\ldots,y_m\in\{0,1\}^n}\langle 0^n|N_m C_m|y_m\rangle \ldots \langle y_2|N_1 C_1|0^n\rangle\right|^2$
  - **Key point:** this is still a polynomial of degree $2m$ in the ideal gate entries
- So by the same arguments as before, we have a worst-to-average case reduction for computing $E_{N\sim\mathcal{N}}[p_{0^n}(N\cdot C)]$ to within $\pm 2^{-n^3}$
  - i.e., if we can compute this quantity for a random C can also compute for a worst case C
  - How hard is that?

Worst-case hardness of computing noisy output probabilities [Fujii ‘16]

- How hard is computing $E_{N\sim\mathcal{N}}[p_{0^n}(N\cdot C)]$ for a worst-case circuit C?
- Fujii has shown this is classically hard if it’s possible to **error detect**
  - i.e., if gate error rate, q, is below a constant error detection threshold
- **Proof idea**
  - As with prior quantum supremacy arguments [BJS’10], it suffices to be able to show universality of noisy quantum circuits under postselection
  - If we can detect errors, we can postselect on the syndrome measurement outcomes corresponding to no error occurring
- This requires high overhead to error detect nearly perfectly
- As a consequence of [Fujii ‘16],[BFNV’19] *computing* output probabilities of noisy *random* quantum circuits is classically hard if the noise per gate is below the **error detection** threshold

New easiness results

- Many recent classical simulation results for restricted classes of random quantum circuits (e.g., [Napp et. al. ’20], [Zhou et al.’20])
- **Our focus: 1D random circuits** with Haar random two-qubit gates and local depolarizing noise
  - **Recall:** With depolarizing noise, output distribution of random circuit eventually converges to uniform
  - But for a given gate error rate, what is the “hardest” depth, system size to implement? Where do quantum correlations “peak”?

Numerical results for noisy 1D RCS [Noh, Jiang, F'20]

- We consider the "*MPO entanglement entropy*" of the resulting mixed state
  - A measure of quantum correlations between two disjoint subsystems of qubits $[1, \dots, \ell], [\ell + 1, \dots, n]$
  - Reduces to standard entanglement entropy in case of pure states
- *Motivation for this quantity:* determines the cost of classical MPO simulation
  - Can compute the output probability in time $\sim 2^{S_{max-MPO-EE}(\rho)}$
    - Because "*Maximum MPO entanglement entropy*" can be used to bound the required bond dimension, χ, needed to accurately describe a mixed state
  - Running time is $poly(n, d, \chi)$ and so exponential in $S_{max-MPO-EE}(\rho)$

Plots from [Noh, Jiang, F'20] (1)

- Each plot has different fixed two-qubit error rate p
- For each system size $n = 4 \dots 18$ we compute the *Max MPO Entanglement Entropy* measure, averaged over N_s =24 different random circuits
- We see that for each error rate, there's a peak depth for which correlations are maximized
- Moreover in each plot, at this peak depth, after sufficiently large system size, adding more qubits doesn't change the *Max MPO Entanglement Entropy*
- So from the perspective of this particular algorithm, once we fix the noise rate, hardness "saturates" at fixed system size.

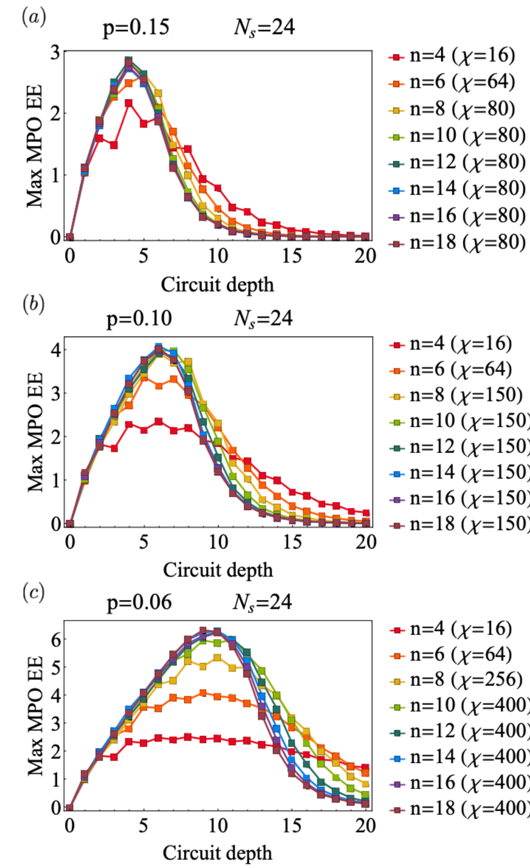


FIG. 5: Maximum MPO entanglement entropy $\mathcal{S}_{\max}$ (averaged over $N_s = 24$ circuit realizations) as a function of the circuit depth D for various number of qubits $4 \leq n \leq 18$ and two-qubit gate error rates (a) $p = 0.15$, (b) $p = 0.1$, and (c) $p = 0.06$. In all cases, we numerically confirm that the chosen bond dimensions are large enough to account for at least 99.1% of the total probability on average.

Plots from [Noh, Jiang, F’20] (2)

- To see this saturation behavior more directly we plot Number of qubits vs *Max MPO Entanglement Entropy*
- Each curve represents a different error-rate at optimal depth for that error-rate (from prior plot)
- Again, we see there’s a maximum system size, determined by the error rate, after which we don’t gain in quantum correlations using this measure

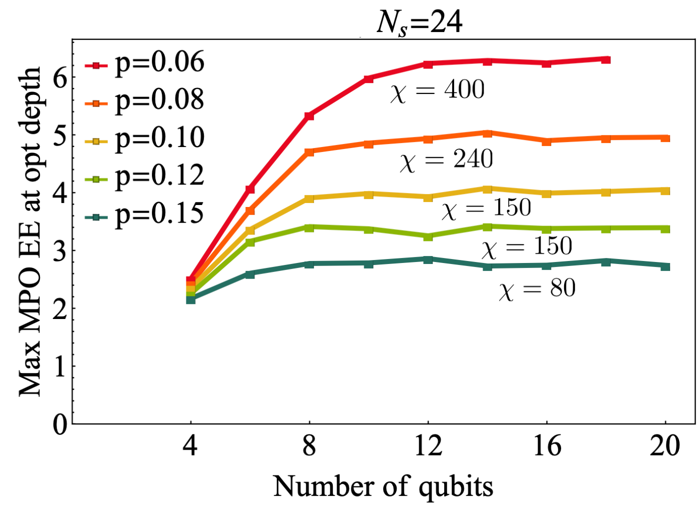


FIG. 8: Maximum achievable MPO entanglement entropy at the optimal circuit depth $\mathcal{S}^{\star}_{\max}$ for various two-qubit gate error rates $0.06 \leq p \leq 0.15$ and number of qubits $4 \leq n \leq 18$. The bond dimension χ used in each case is specified next to each curve.

Conclusions

- Numerically, we observe that for noisy 1D random circuits there is a sense in which quantum correlations peak at a particular system size
- We can make use of this observation to compute noisy output probabilities using Matrix Product Operator (MPO) methods
- On the other hand, we can prove that computing noisy output probabilities (to extreme precision) is hard in 2D below a noise threshold
  - combining our results from [BFNV'19] with [Fujii'16]

Thanks!

Thanks also for many helpful discussions on related topics:
Abhinav Deshpande [Maryland]
Adam Bouland, Yunchao Liu, Umesh Vazirani [Berkeley]
Dorit Aharonov [Hebrew University]
Roozbeh Bassirianjahromi, Liang Jiang, Kyungjoo Noh [UChicago]
Jens Eisert and group [Berlin]