Google AI
Quantum

Quantum Supremacy using a Programmable Superconducting Processor

Sergio Boixo

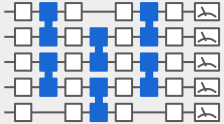
May 4th, 2020


Google AI
Quantum
Google

Demonstrating quantum supremacy

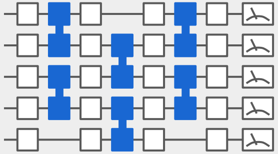
Task

Sample the output distribution of a random quantum circuit.

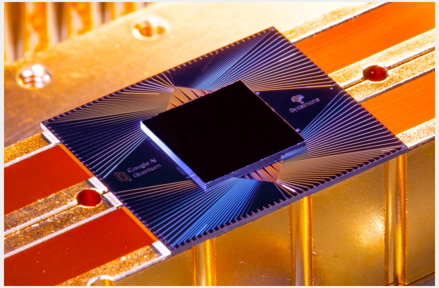


Quantum vs. Classical

Best quantum strategy:



Run the circuit on a **quantum processor**



Best classical strategy:

Simulate (sample) the circuit using a **supercomputer**

Evaluation

Quantum processing time vs. Classical processing time

Short — Unfeasible in practice

“Quantum supremacy”

Google AI Quantum

Sampling vs. estimating probabilities

- For random quantum circuits, we only know how to sample the output by estimating probabilities.
  - Computational cost (2D) proportional to fidelity (Markov et. al. 1807.10749)

- A polynomial classical sampling algorithm implies that probabilities can be estimated using an NP oracle.
  - Count the number of assignments to the “random bits” for a given output using the NP oracle (Stockmeyer's Approximate Counting 83, Aaronson & Arkhipov 2010).
  - This holds for (polynomial) globally unbiased noise (Google and Brandao Nat. 2019).

- Sampling with a quantum circuit does not imply estimating probabilities (even with an NP oracle).

Random circuit sampling (RCS) complexity

- Estimating an output probability for quantum circuits is known to be worst case #P.

- Calculating output probabilities of random quantum circuits "exactly" is proven average case #P (Bouland et. al. Nat. Phys. 2019, Movassagh 1909.06210).

- Proving average case estimation hardness of output probabilities would imply RCS is hard (see Eisert next talk).

- RCS conjectured to be hard (Boixo et. al. Nat. Phys. 2018, Aaronson & Chen 2016, Bremner et. al. 2016).

- Without error correction, fidelity decays exponentially (in #gates), and asymptotic complexity theory does not apply.

XE benchmarking (fidelity estimation)

- Random circuits (Emerson et. al., RB) and quantum chaos (with V. Smelyanskiy). Sensitivity to errors related to computational hardness.

- System fidelity: is everything working?

- XEB requires simulating the system. As fidelity improves for multiqubit systems, the simulation cost grows generally exponentially. Hence the name quantum supremacy for this experiment.
  - But note that patch XEB works fine.

Boixo et. al. 2016 Nat. Phys 2018, C. Neill Science 2018.

XEB for large systems

Output of noisy random circuit is $\rho_U = p|\psi_U\rangle\langle\psi_U| + (1-p)\chi_U$

p like depolarization fidelity, $|\psi_U\rangle$ is the ideal output, and χ_U is the result of errors.

We make an observable which is a function of the ideal simulated probabilities $p_s(q) = |\langle q|\psi_U\rangle|^2$
Measure bitstring q and map it to real number $f(p_s(q))$:

$$O_U = \sum_q f(p_s(q))\,|q\rangle\langle q|$$

Assume that the noisy operator χ_U from a random quantum circuit results in probabilities $\langle q|\,\chi_U\,|q\rangle$ with average $1/D = 2^{-n}$ uncorrelated with $f(p_s(q))$. By the central limit theorem:

$$\mathrm{Tr}\, O_U \chi_U = \sum_q \langle q|\,\chi_U\,|q\rangle\, f(p_s(q)) = \frac{1}{D}\sum_q f(p_s(q)) + O\left(\frac{1}{\sqrt{D}}\right)$$

Central limit theorem

XEB and concentration of measure (large systems)

Output of noisy random circuit is $\rho_U = p|\psi_U\rangle\langle\psi_U| + (1-p)\chi_U$

By concentration of measure for random circuit U and observable O_U expect for random χ_U:

$$\mathrm{Tr}\, O_U \chi_U = \frac{\mathrm{Tr}\, O_U}{D} + O\left(\frac{1}{\sqrt{D}}\right)$$

“A function on a random point of a high dim. space (sphere) concentrates on (returns) the average value.”

We make an observable which is a function of the ideal simulated probabilities $p_s(q)$ to avoid concentration for the ideal output $|\psi_U\rangle$

For simplicity, we might write $\chi_U = 1/D$. But it also works for coherent errors and approximate classical simulations.

Estimating fidelity with XEB

From $\rho_U = p|\psi_U\rangle\langle\psi_U| + (1-p)\chi_U$

and observable $O_U = \sum_q f(p_s(q))|q\rangle\langle q|$ ($p_s(q)$ is ideal simulated prob. of q)

using concentration of measure $\mathrm{Tr}\, O_U \chi_U = \frac{\mathrm{Tr}\, O_U}{D} + O\left(\frac{1}{\sqrt{D}}\right)$

we solve for fidelity $\langle O_U \rangle_\rho = p\langle O_U \rangle_U + (1-p)\frac{\mathrm{Tr}\, O_U}{D}$

Measure

Estimated

Simulation or analytics

XEB observable or p (fidelity) estimator

Measure bitstrings {q} with probabilities $p\, p_s(q) + (1-p) \langle q | \chi_U | q \rangle$
where χ_U is the result of (generic) errors and $p_s(q)$ are ideal simulated probabilities.

From concentration of measure:
Cross entropy (XE) estimator $\overline{\langle \log p_s(q) \rangle} \simeq p - \log D - \gamma$

Or linear XEB $p \simeq \overline{\langle D p_s(q) - 1 \rangle}$ (max. likelihood)

Linear XEB has lower variance (with V. Smelyanskiy and A. Korotkov).
Also normalized prob. of “heavy” output (Aaronson & Chen, quantum volume).
Using the same data and simulation, try several estimators and check they agree.

Numerical checks for concentration of measure

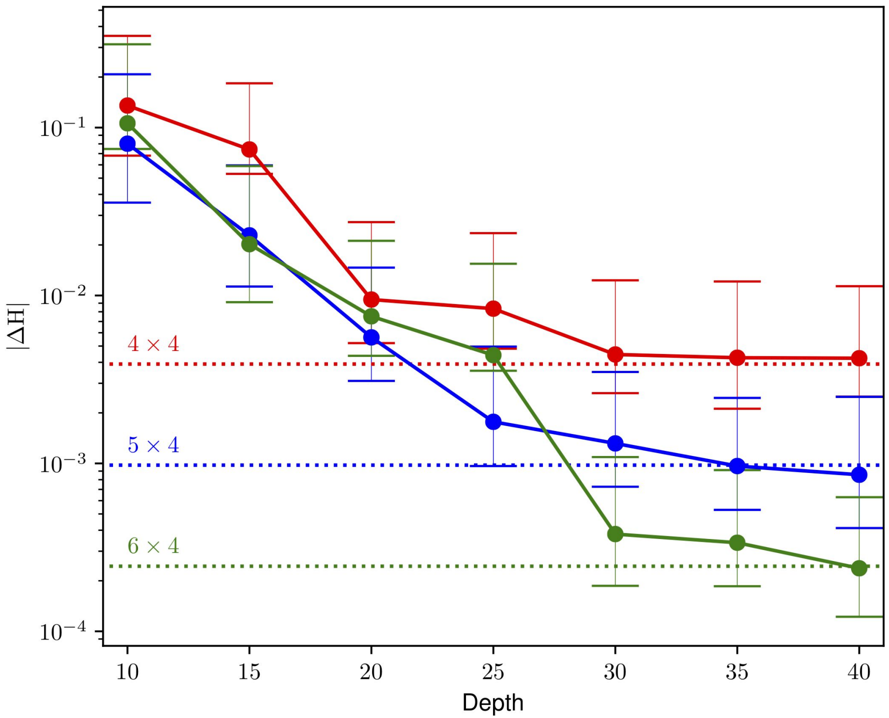
We obtain numerically that after a single Pauli error

$$\left| \langle O \rangle_{\text{Pauli error}} - \frac{\operatorname{Tr} O}{D} \right| \simeq \frac{1}{\sqrt{D}}$$

Also checks for 2D pure states with approximate sampling algorithms:

Markov et. al. arXiv:1807.10749,
Villalonga et. al. arXiv:1811.09599

1D MPS? (2002.07730)



Boixo et. al. Nat. Phys. 2018

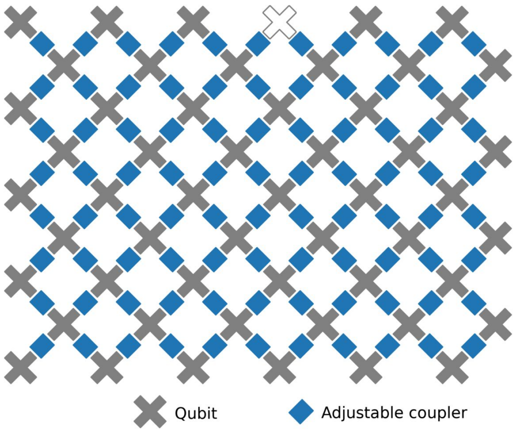
Google AI Quantum

XEB algorithm (Quantum supremacy experiment)

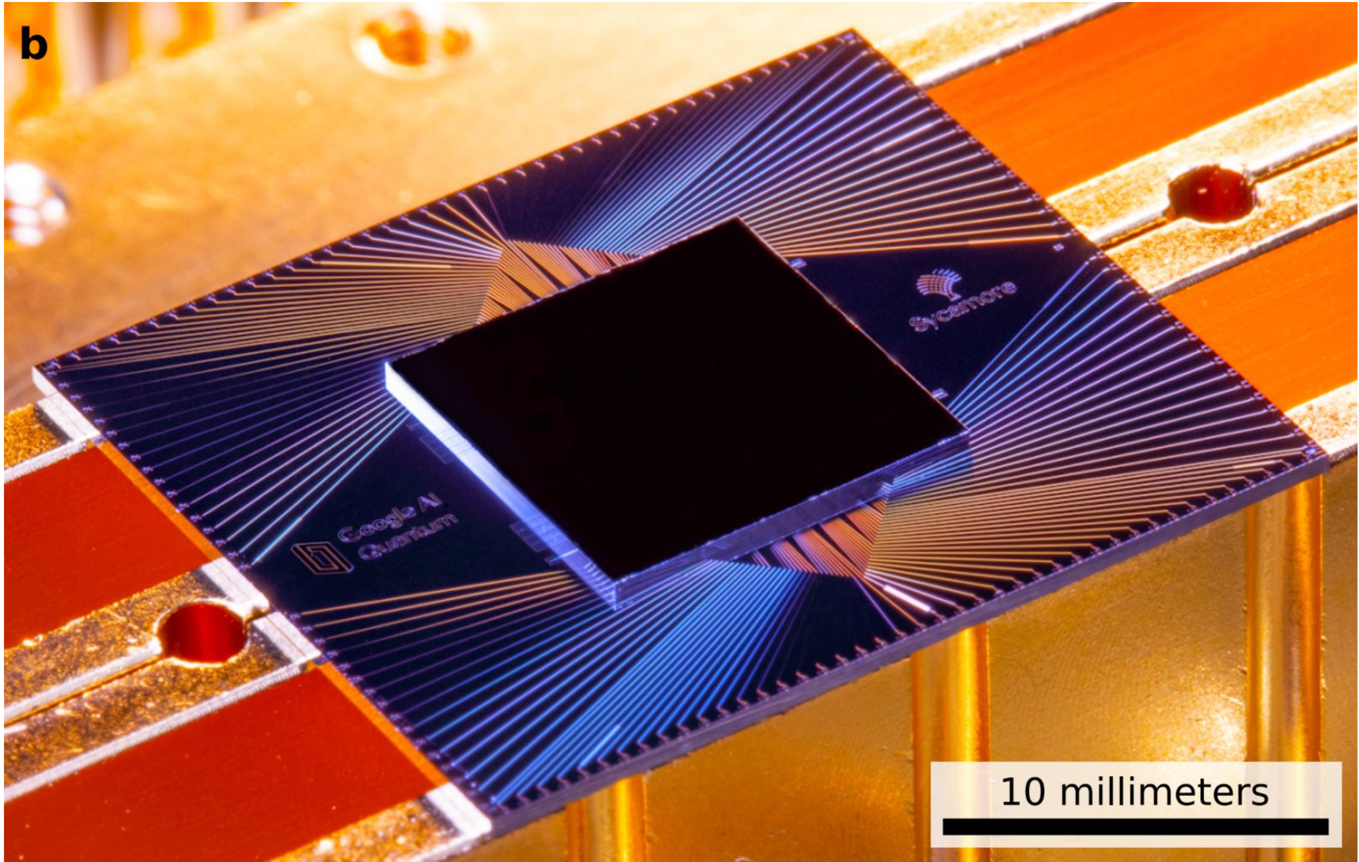
Boixo et. al. Nat. Phys. 2018

```
circuit = random_circuit()

sample_count = 10**6

# Run on real hardware.
actual_samples = sample_circuit(circuit, sample_count)

# Determine the ideal probability of the sample.
p = 1
for s in actual_samples:
  # EXPENSIVE!
  p *= simulate_ideal_probability(circuit, s)

# Derive cross entropy from ideal probability.
cross_entropy = -log(p) / sample_count
```

Sycamore Processor: 54 qubits, 196 control knobs

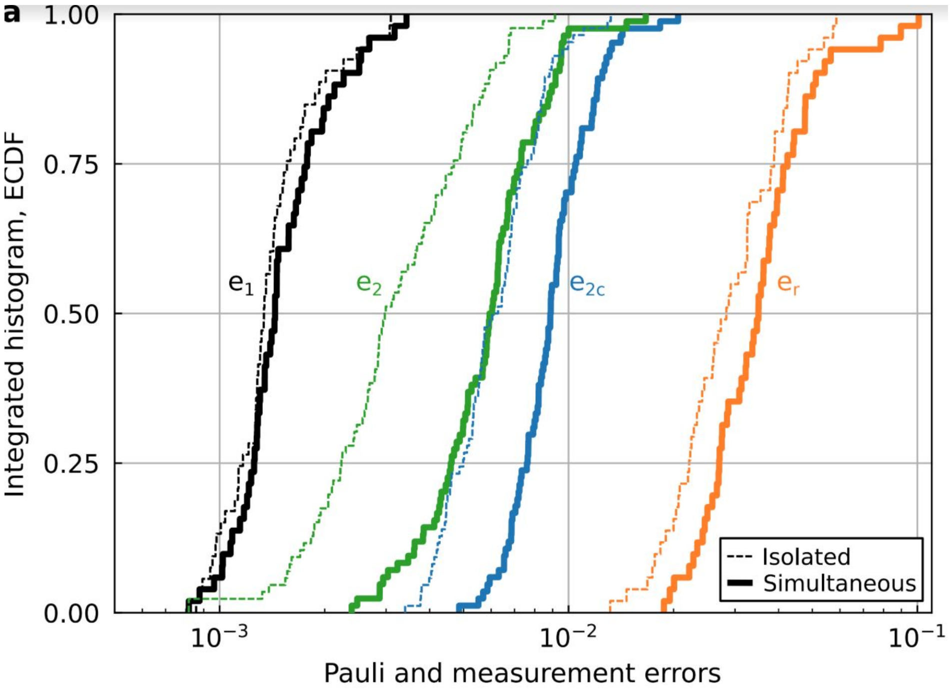


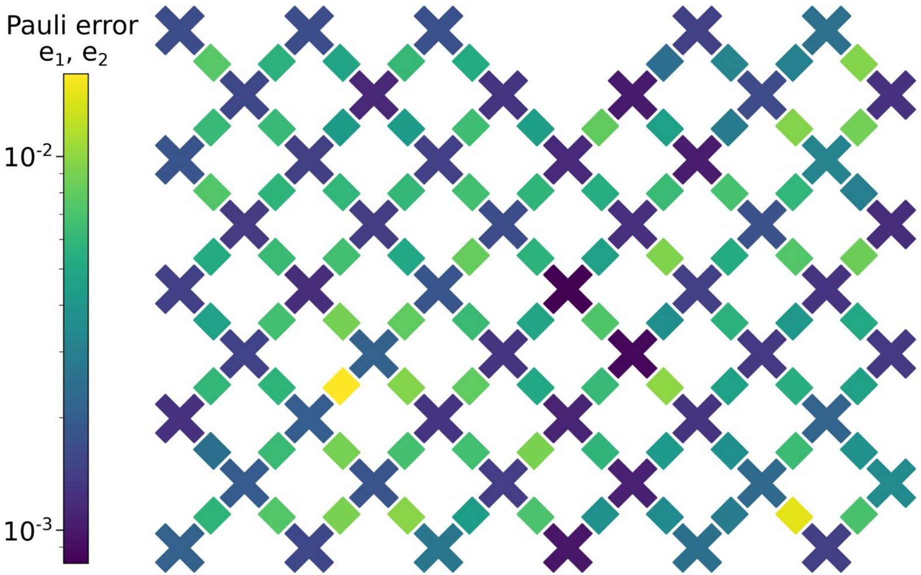
New



Google AI Quantum

Low Errors using Fast 2-Qubit Gates (12 ns)



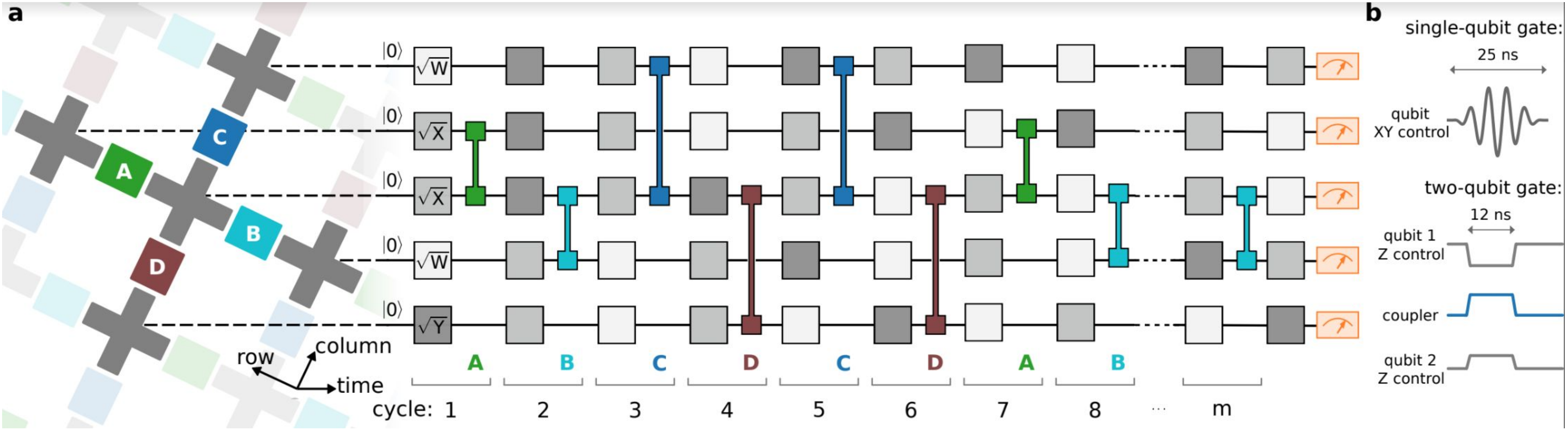


Average error	Isolated	Simultaneous
Single-qubit (e_1)	0.15%	0.16%
Two-qubit (e_2)	0.36%	0.62%
Two-qubit, cycle (e_{2c})	0.65%	0.93%
Readout (e_r)	3.1%	3.8%

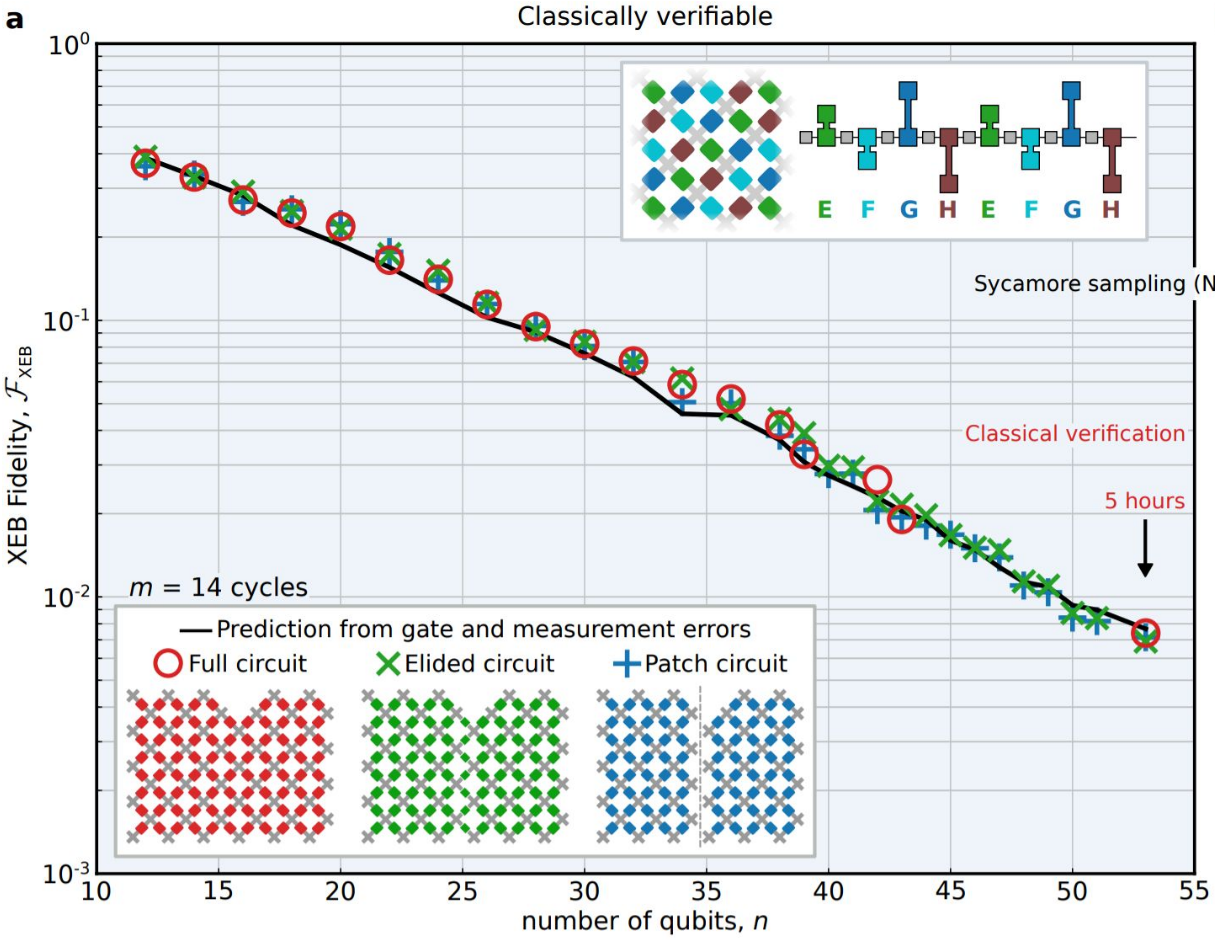
Need to quote
Average and **Simultaneous**

Control Sequence for Quantum Supremacy

- Simultaneous gates all qubits
- General purpose algorithm
- 1 million outputs ~ 200 seconds



Quantum Supremacy Data



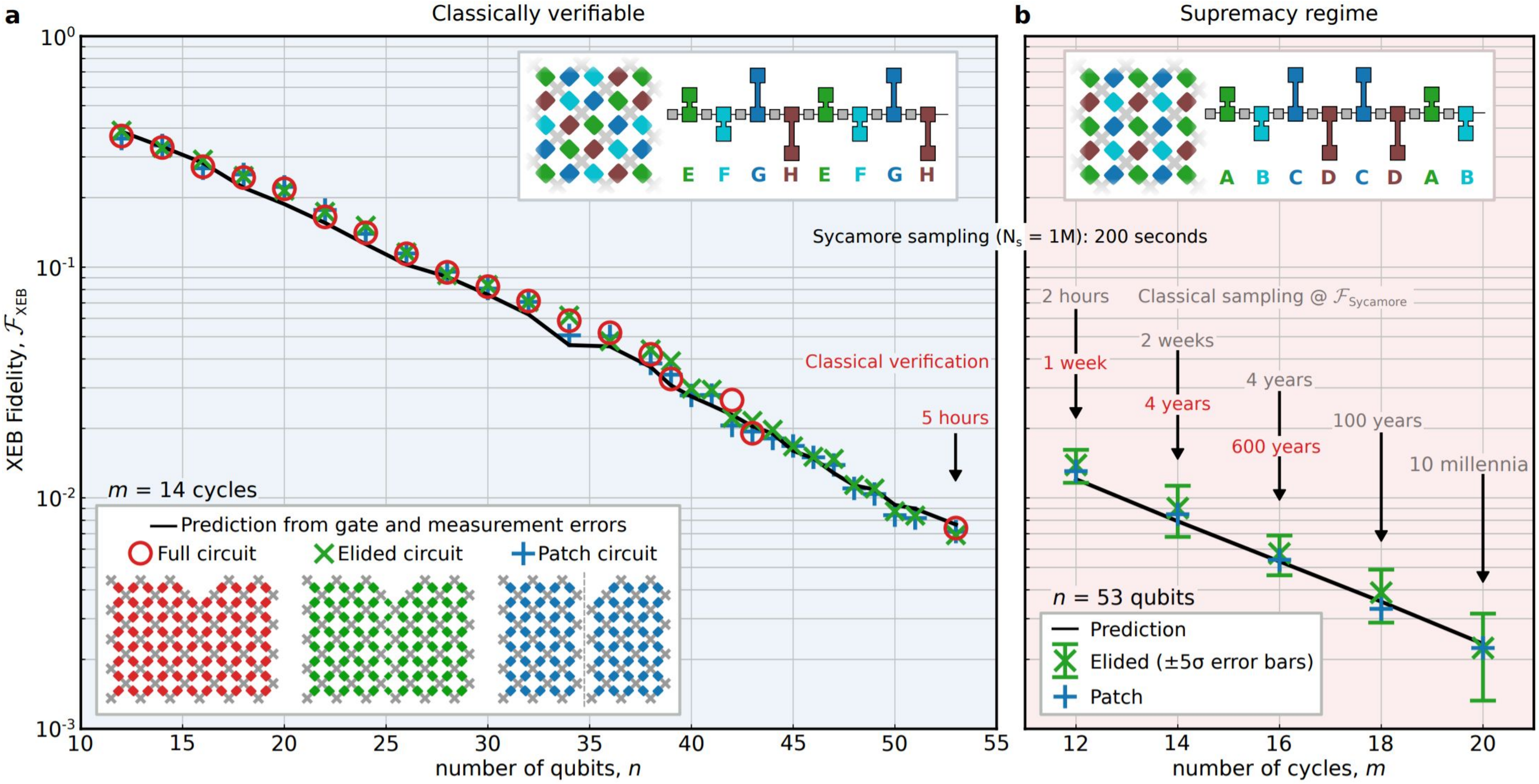
Black line is a discrete error model:

$$\Pi_i (1-e_i)$$

Works because:

1. Random circuits randomize systematic errors.
2. Crosstalk here is small and local
3. Simultaneous error measurement includes crosstalk.

Quantum Supremacy Data



Schrodinger-Feynman Hybrid simulation

https://github.com/quantumlib/qsim

Divide the quantum circuit in two patches, left and right.

Use the Schmidt decomposition of cross gates which link the patches:

$$U = \frac{1}{2} \sum_k w_k V_k \otimes W_k$$

With g cross gates, output amplitudes require (Schmidt rank)^g simulations

$$\langle x|\psi\rangle = 2^{-g} \sum_{k_1,\ldots,k_g} w_{k_1} \cdots w_{k_g} \langle x^l|\psi^l_{k_1,\ldots,k_g}\rangle \langle x^r|\psi^r_{k_1,\ldots,k_g}\rangle$$

We obtain amplitudes with fidelity F with cost proportional to F:

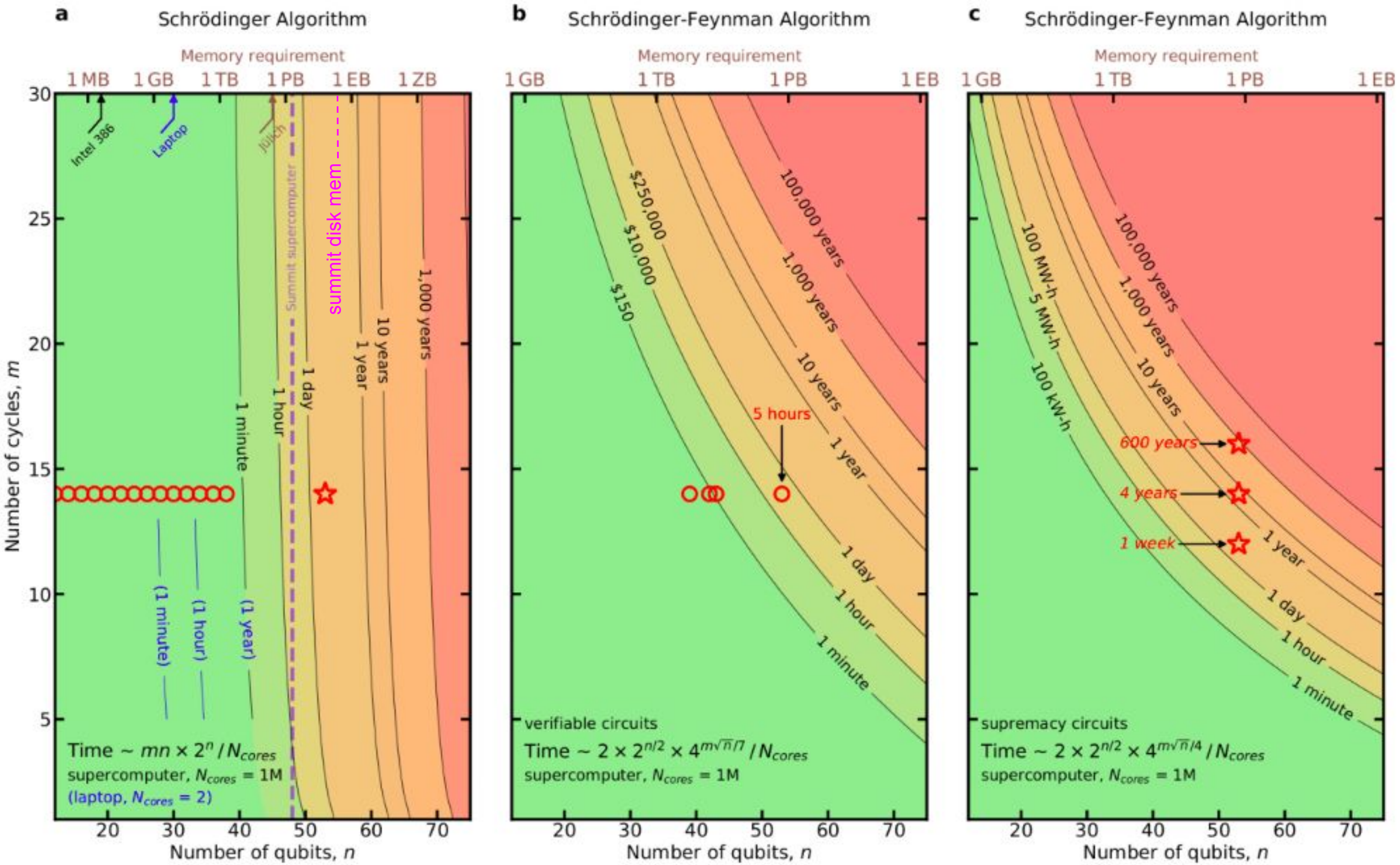
summing a fraction F of paths (if equal magnitude Schmidt coefficients).

Note: there are efficient 1D (quite) noisy simulations (2002.07730, 2003.13163)

Use rejection sampling (~10 amplitudes per output bitstring).

Markov, Fatima, Isakov, Boixo (2018), Aaronson & Chen (2016).

Simulation Cost



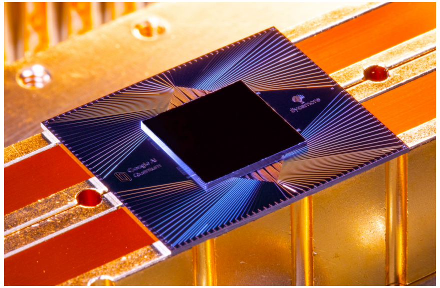
Goog

Improving Computer Simulation

- “We expect that lower simulation costs than reported here will eventually be achieved, but we also expect that they will be consistently outpaced by hardware improvements on larger quantum processors.”

- Strongly support **running** validation programs
  - Tricky to write efficient supercomputer code, failures
  - Untested proposal for disk memory use, possible?
  - All experimental data posted for checking

- There will be a 57+ qubit Sycamore processor



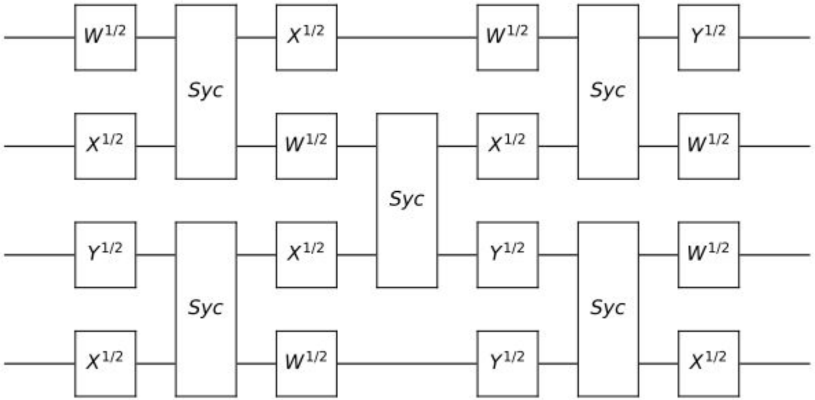
Feynman algorithm (tensor networks)

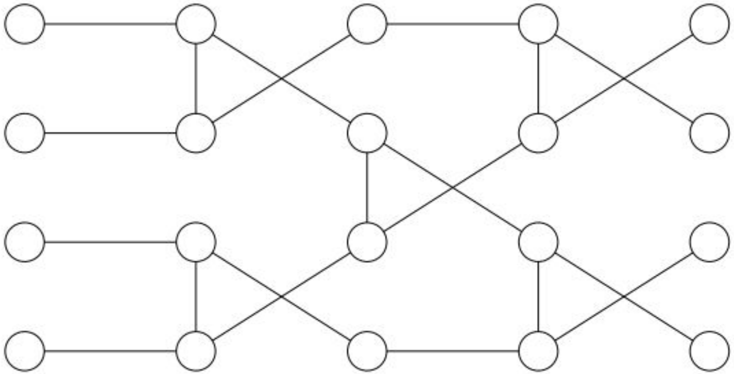
Markov and Shi 2008, Boixo et. al. 2016, Aaronson and Chen 2016, Boixo et. al. 2017, Alibaba ..., J. Gray

- A quantum circuit can be mapped to an undirected graphical model (or complex Ising model).

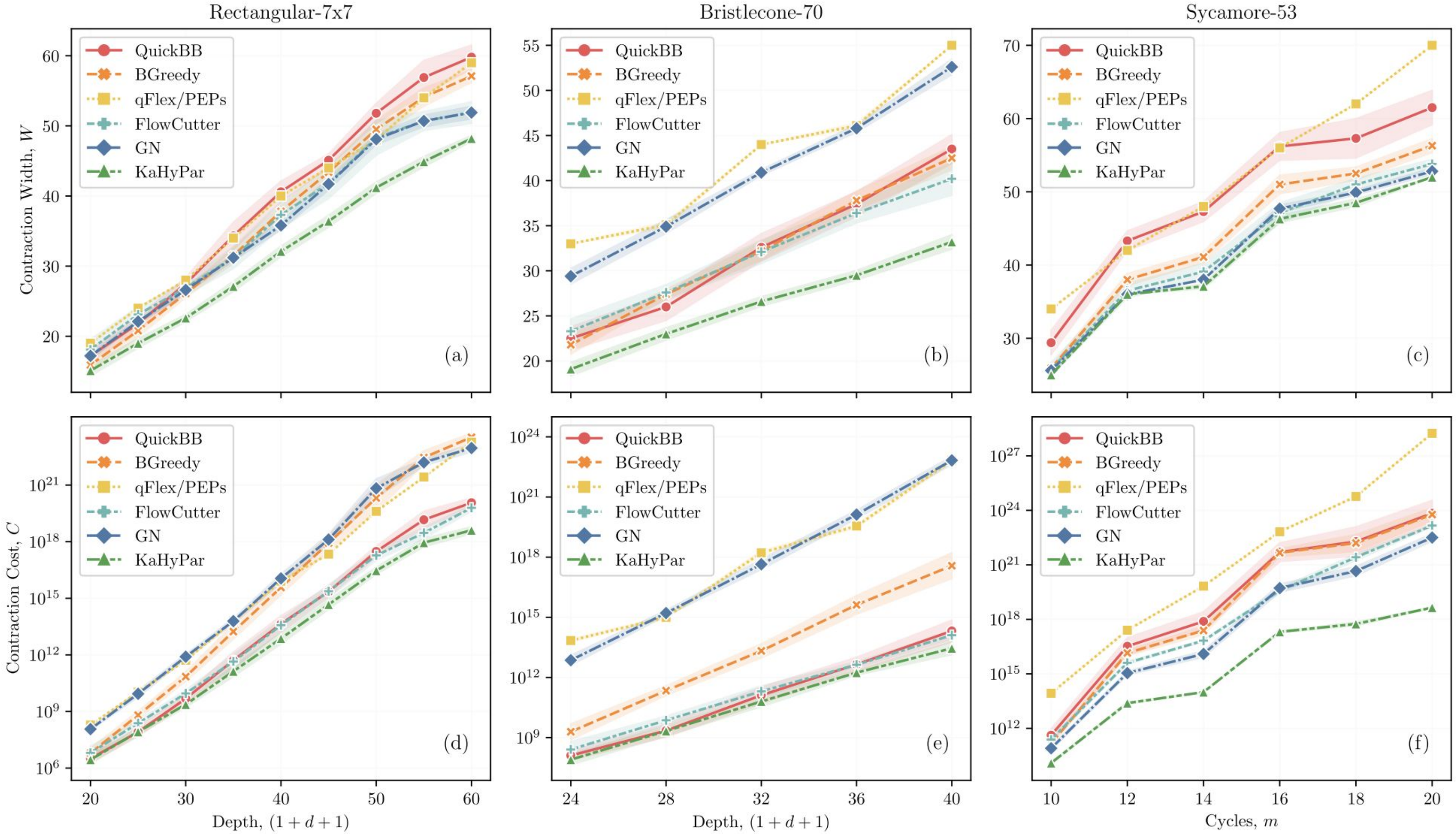
- The cost of computing one output probability is (up to a small constant) given by the treewidth.

- The treewidth for a 2D circuit scales like exp(min(d√n, n), for depth d. (Think of tensor contraction).





Improved contraction orderings. J. Gray and S. Kourties 2020



Latest timings with FA (TN) J. Gray and S. Kourties 2020

Circuit	time (sec)	C_s	Slicing Overhead (C_s/C_{best})	d_{sliced}	FLOPs Efficiency
Sycamore-53* (m=12)	7.87×10^2	2.42×10^{13}	1.67×	2^9	**8.16%**
Sycamore-53* (m=14)	$^*2.92 \times 10^3$	2.53×10^{14}	2.63×	2^{12}	22.9%
Sycamore-53* (m=16)	$^*3.01 \times 10^6$	3.43×10^{17}	7.43×	2^{22}	30.1%
Sycamore-53* (m=18)	$^*2.66 \times 10^7$	3.62×10^{18}	11.3×	2^{24}	36.0%
Sycamore-53* (m=20)	$^*7.17 \times 10^9$	1.50×10^{21}	**431**×	2^{32}	55.3%

- One approximate amplitude (0.5 fidelity) at 7 billion seconds in one GPU
  - That's 3 days in Summit (28K GPUs), but can be faster (better GPUs). 4 hours? Doable.
  - One exact amplitude not doable.

- ~ 3 Summit years for 3M amplitudes at 0.2% fidelity

- ~ 1000 Summit years for XEB (3 million exact amplitudes).

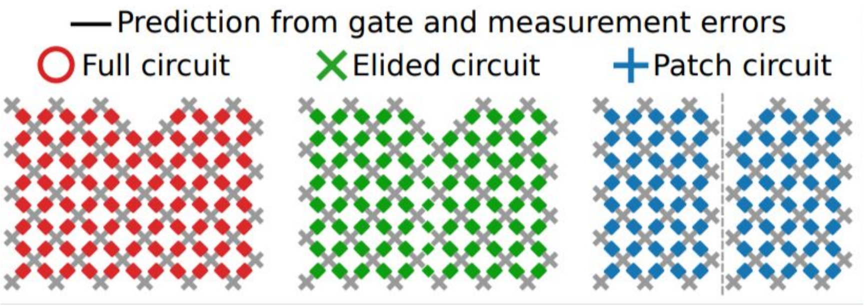
Quantum Science Results

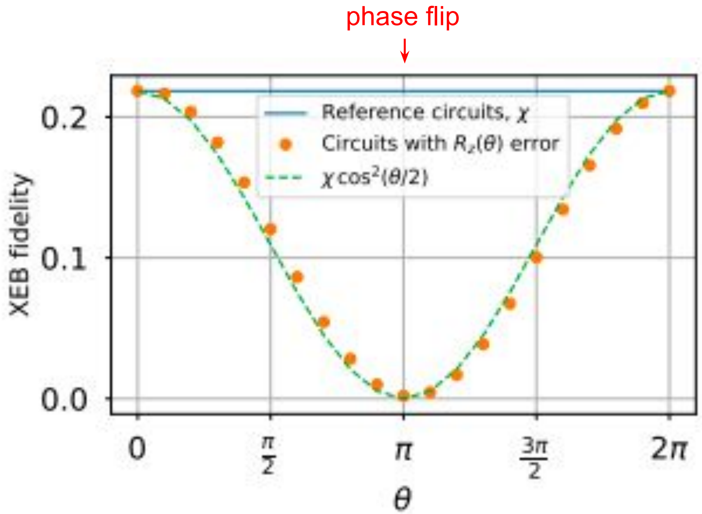
1) Same fidelity: full, elided, patch, predicted
   Errors NOT depend on entanglement
   and computation complexity!

2) No new decoherence physics:
   Probability prediction, Fidelity = $\Pi_i (1-e_i)$
   Error correction should work

3) Quantum works at $2^{53} = 10^{16}$ Hilbert space
   Previously tested to $\sim 10^3$

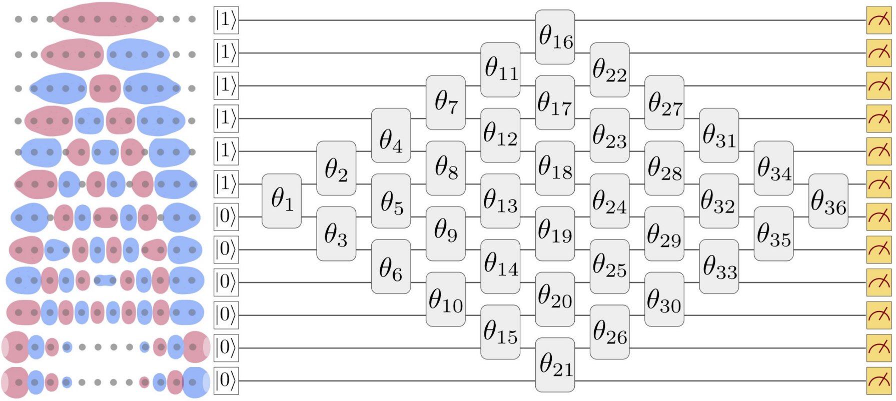
4) Challenge the Extended Church-Turing thesis.

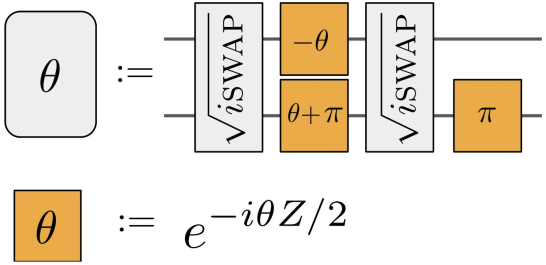




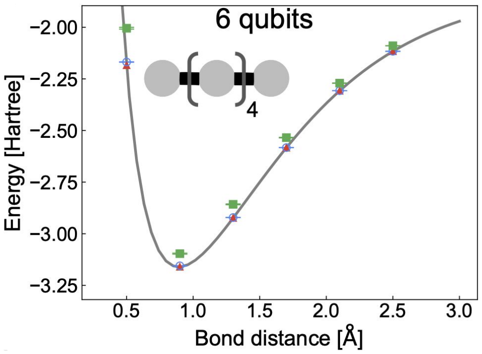
Google AI Quantum

Hydrogen chain to benchmark Sycamore (2004.04174)

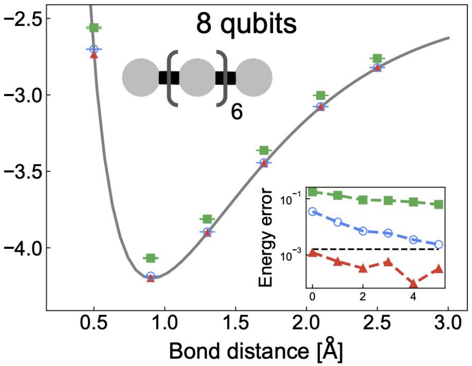




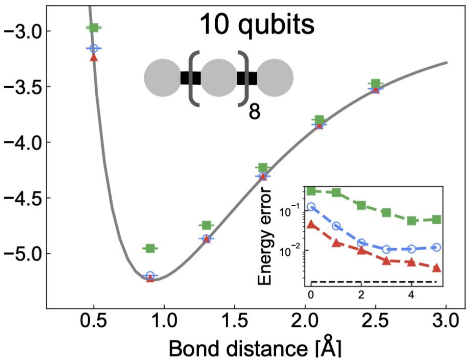
18 sqrt(iswap), 27 Rz



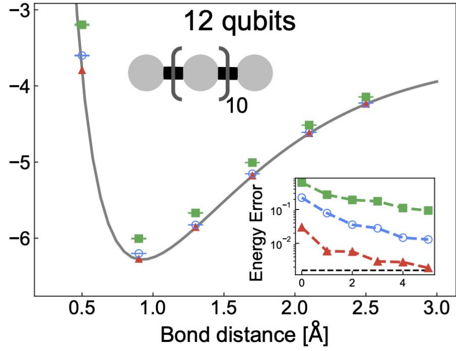
32 sqrt(iswap), 48 Rz



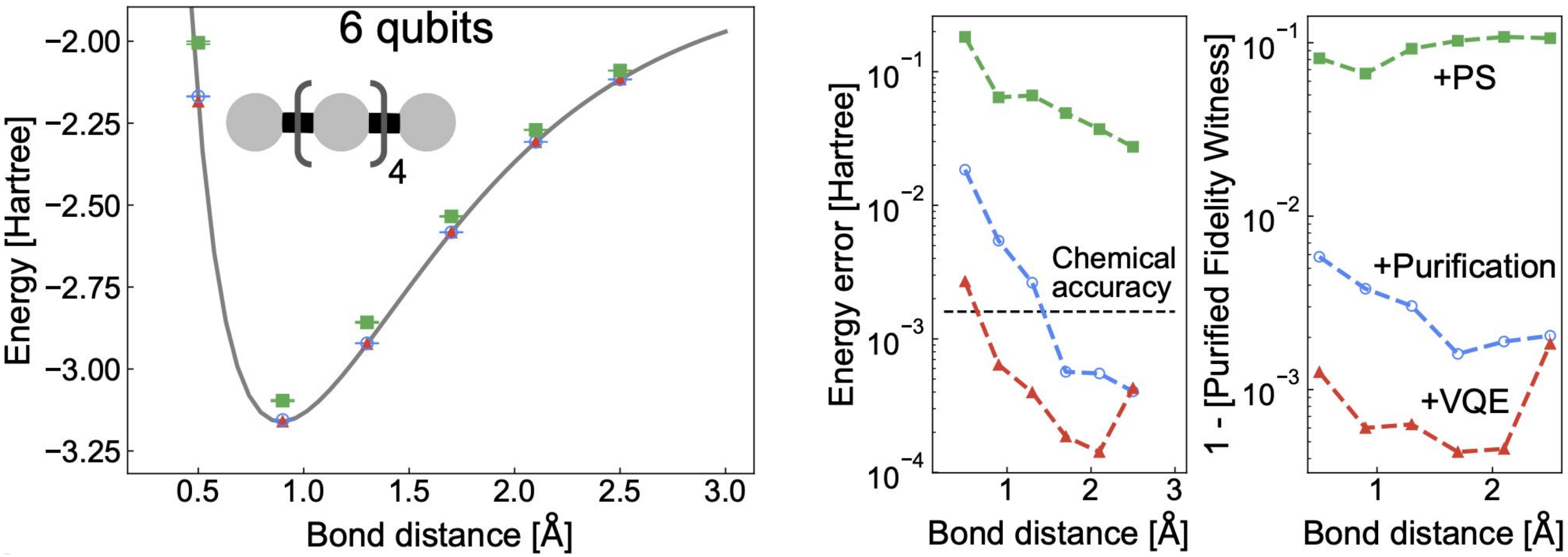
50 sqrt(iswap), 60 Rz



72 sqrt(iswap), 108 Rz



Hydrogen chain to benchmark out device



system	raw	+post-selection	+purification	+VQE
H_6	0.674(2)	0.906(2)	0.9969(1)	0.99910(9)
H_8	0.464(2)	0.827(2)	0.9879(3)	0.99911(8)
H_{10}	0.316(2)	0.784(3)	0.9704(5)	0.9834(4)
H_{12}	0.010(2)	0.654(3)	0.9424(9)	0.9913(3)

Thanks!