# A look into the quantum supremacy experiment

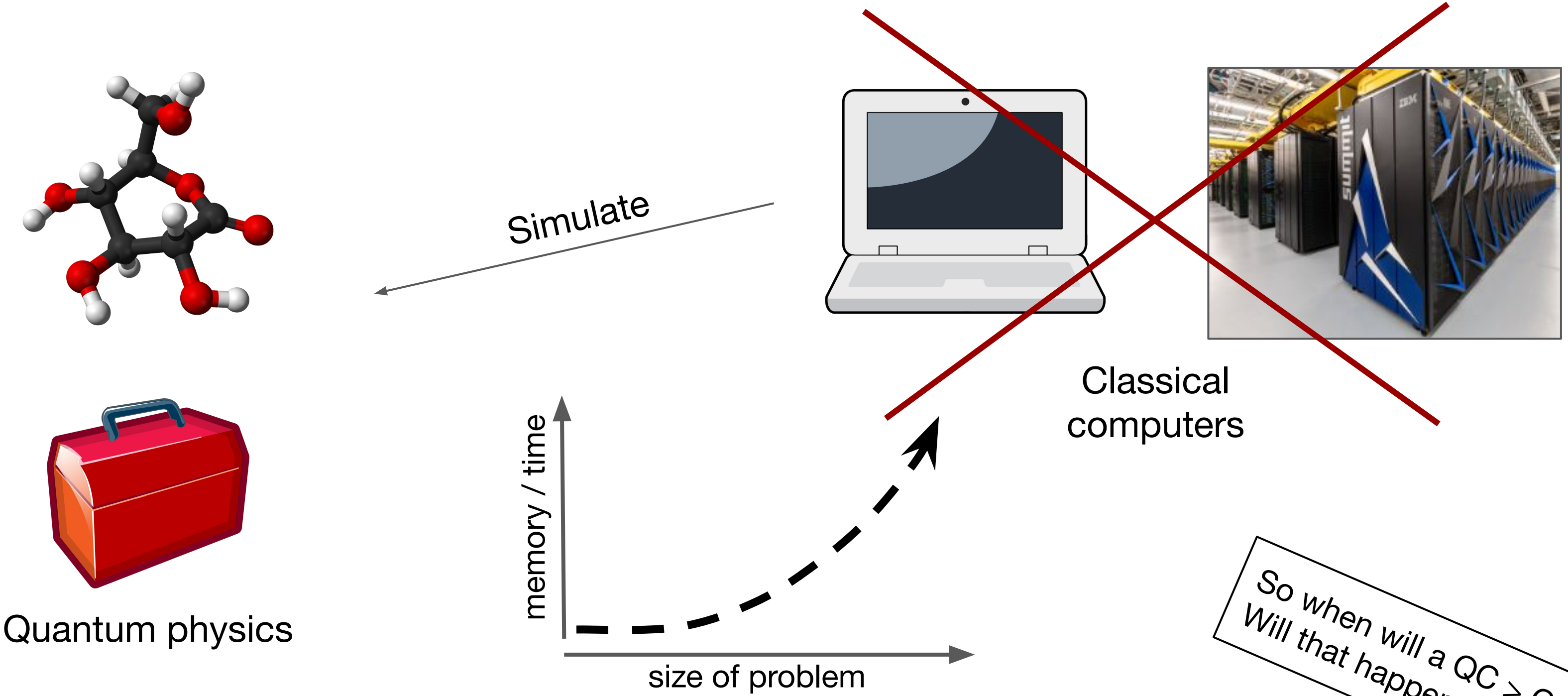
Benjamin Villalonga
(Durham, 08-27-2020)

ILLINOIS

Google AI
Quantum

# PART 1
## Overview

# The promise of quantum computers



Simulate

Quantum physics

Classical computers

So when will a QC > CC in practice?
Will that happen at all? (Noise)

Classical computers are very ***inefficient*** in simulating quantum systems!

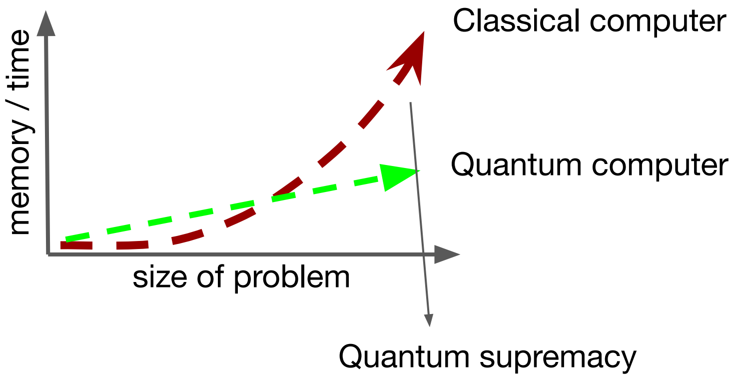
Can we use quantum systems to process information in a fundamentally new way?

# What is quantum supremacy?

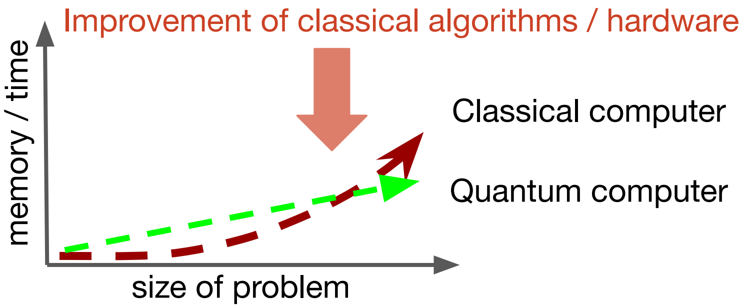
# Proof of principle that quantum computers work

Problem that is:

- Hard for a classical computer
- Easy for a quantum computer
- Correct solution can be "verified"

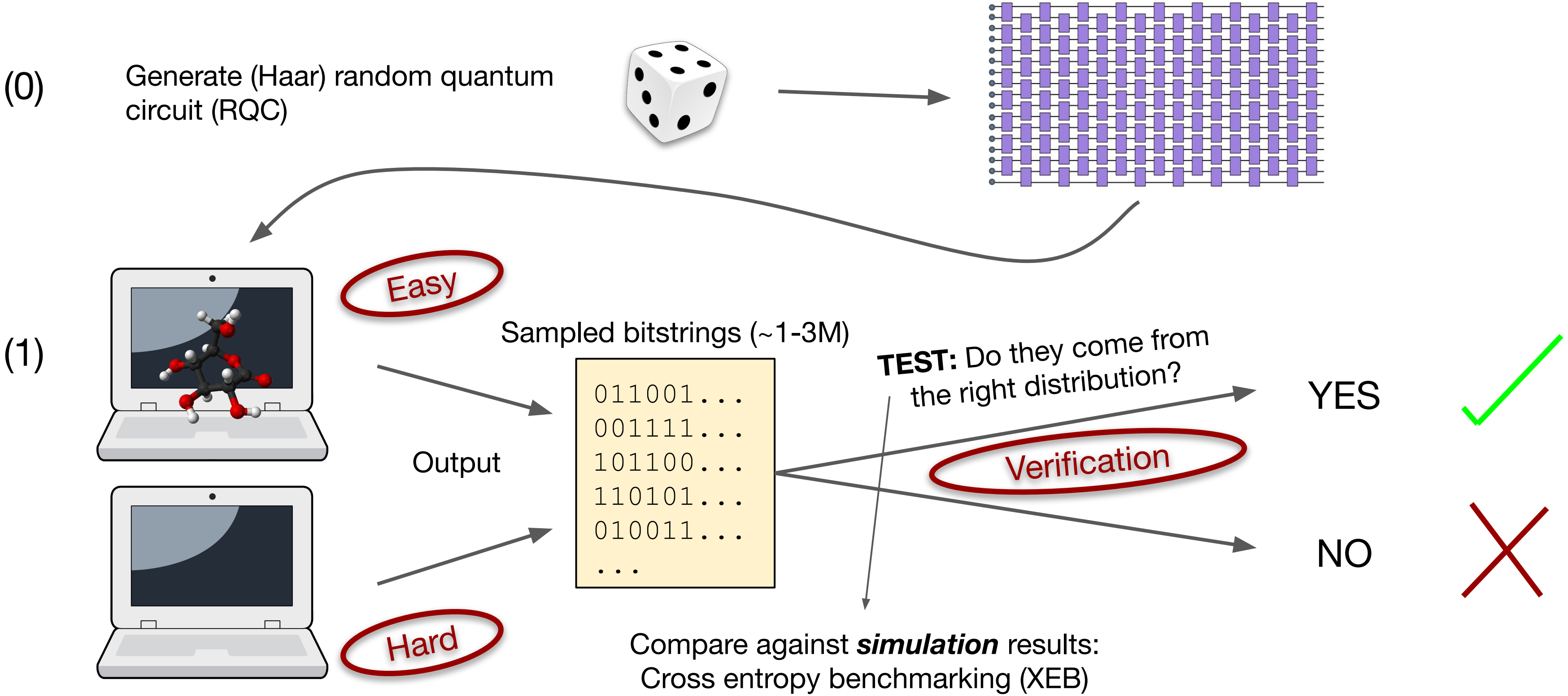


## Does it have to be useful? **NO**



# The problem: Random circuit sampling

(Sergio Boixo)



(0) Generate (Haar) random quantum circuit (RQC)

(1)

Easy

Hard

Output

Sampled bitstrings (~1-3M)

```
011001...
001111...
101100...
110101...
010011...
...
```

**TEST:** Do they come from the right distribution?

Verification

YES ✓

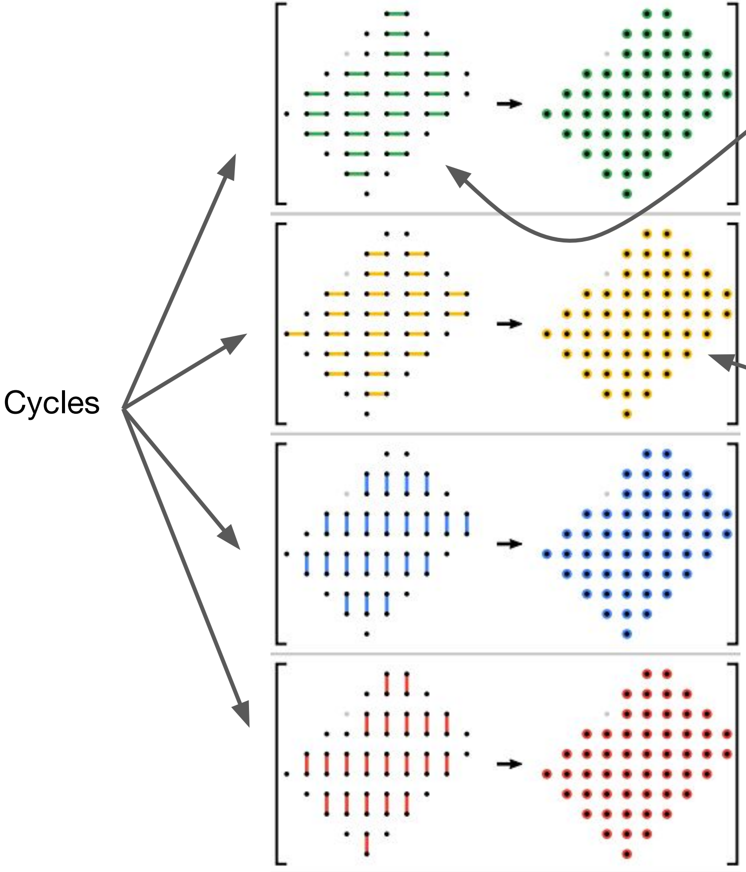
NO ✗

Compare against ***simulation*** results: Cross entropy benchmarking (XEB)

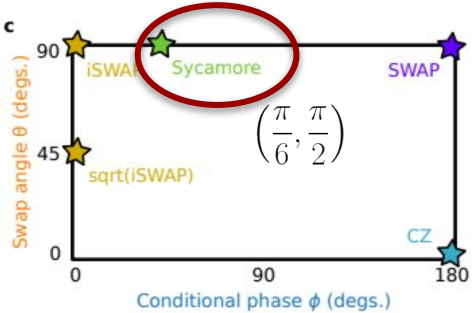
**Characterizing quantum supremacy in near-term devices**

Sergio Boixo[1*], Sergei V. Isakov[2], Vadim N. Smelyanskiy[1], Ryan Babbush[1], Nan Ding[1], Zhang Jiang[3,4], Michael J. Bremner[5], John M. Martinis[6,7] and Hartmut Neven[1]

# ***Generating*** a random quantum circuit



$$\mathrm{fSim}(\theta,\phi)=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos(\theta) & -i\sin(\theta) & 0\\ 0 & -i\sin(\theta) & \cos(\theta) & 0\\ 0 & 0 & 0 & e^{-i\phi}\end{bmatrix}$$

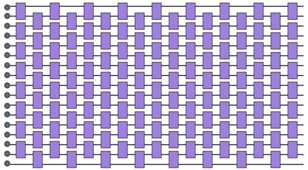


$$\left(\frac{\pi}{6},\frac{\pi}{2}\right)$$

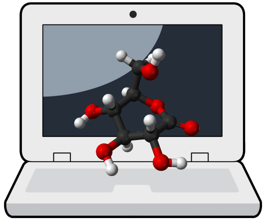
$$\left\{\begin{aligned} X^{1/2} &\equiv R_X(\pi/2)=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & -i\\ -i & 1\end{bmatrix},\\ Y^{1/2} &\equiv R_Y(\pi/2)=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & -1\\ 1 & 1\end{bmatrix},\\ W^{1/2} &\equiv R_{X+Y}(\pi/2)=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & -\sqrt{i}\\ \sqrt{-i} & 1\end{bmatrix}\end{aligned}\right.$$

Discretized Haar RQC

# ***Sampling*** from a random quantum circuit | ***Verifying***



$$= |\psi\rangle = \sum_j c_j |j\rangle$$



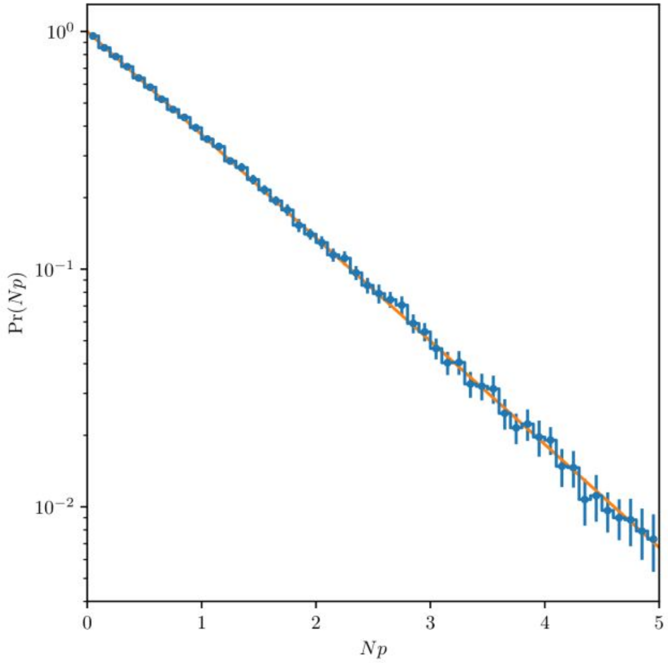
```
011001...
001111...
101100...
110101...
010011...
...
```

$p_1 = |c_1|^2$
$p_2 = |c_2|^2$
$p_3 = |c_3|^2$
$p_4 = |c_4|^2$
$p_5 = |c_5|^2$
...

Sample small number
(1-3M out of $2^{53}$)

Classically compute

**Porter-Thomas** distribution

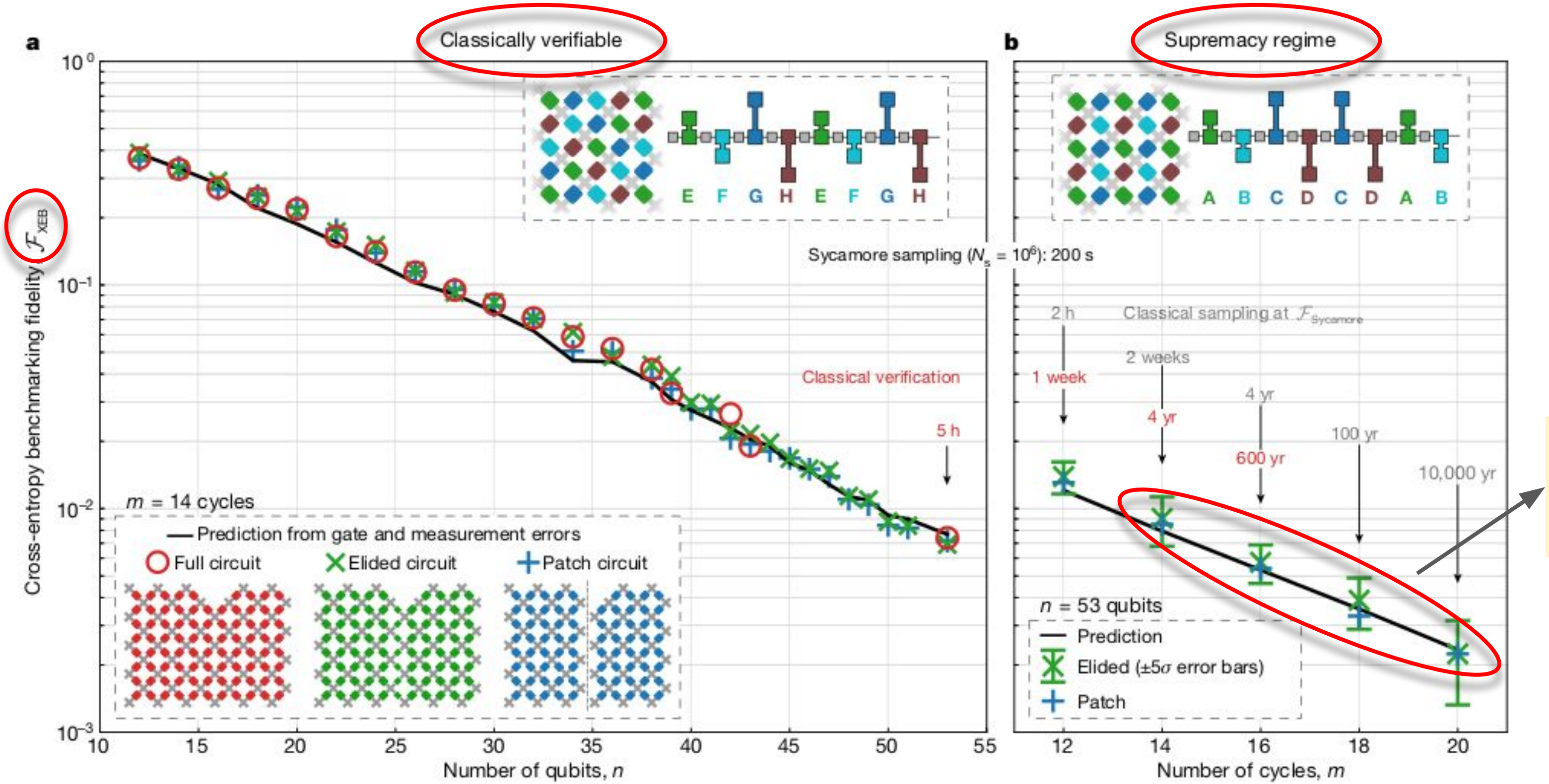


$$Pr(p) \propto e^{-Np}$$

Cross entropy benchmarking (XEB):

$$F = \langle Np(s) - 1 \rangle_{\text{sampled bitstrings}}$$

A single Pauli error gives F~0.



Not verifiable,
*but archived*

**Article**

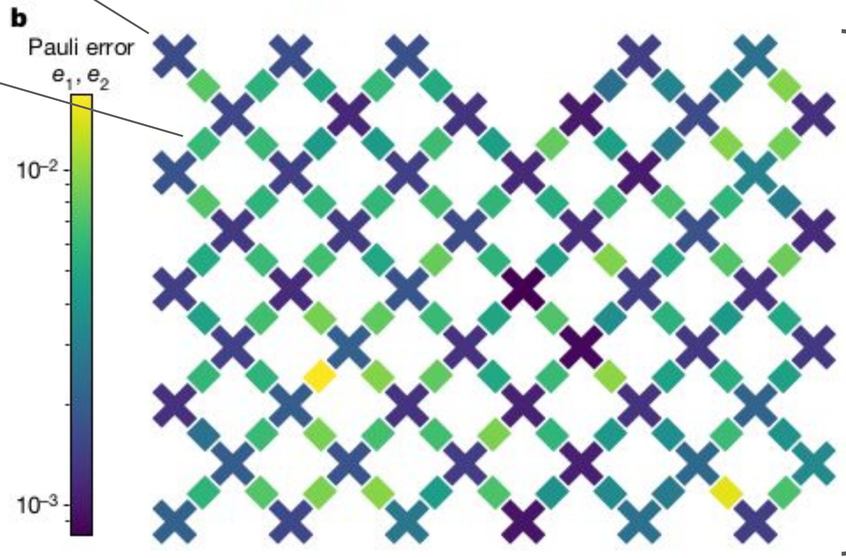
# Quantum supremacy using a programmable superconducting processor

# Brief hardware overview

(John Martinis)

# The need for many **qubits**, **depth**, **connectivity**, and **fidelity**

| 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% |



Qubit

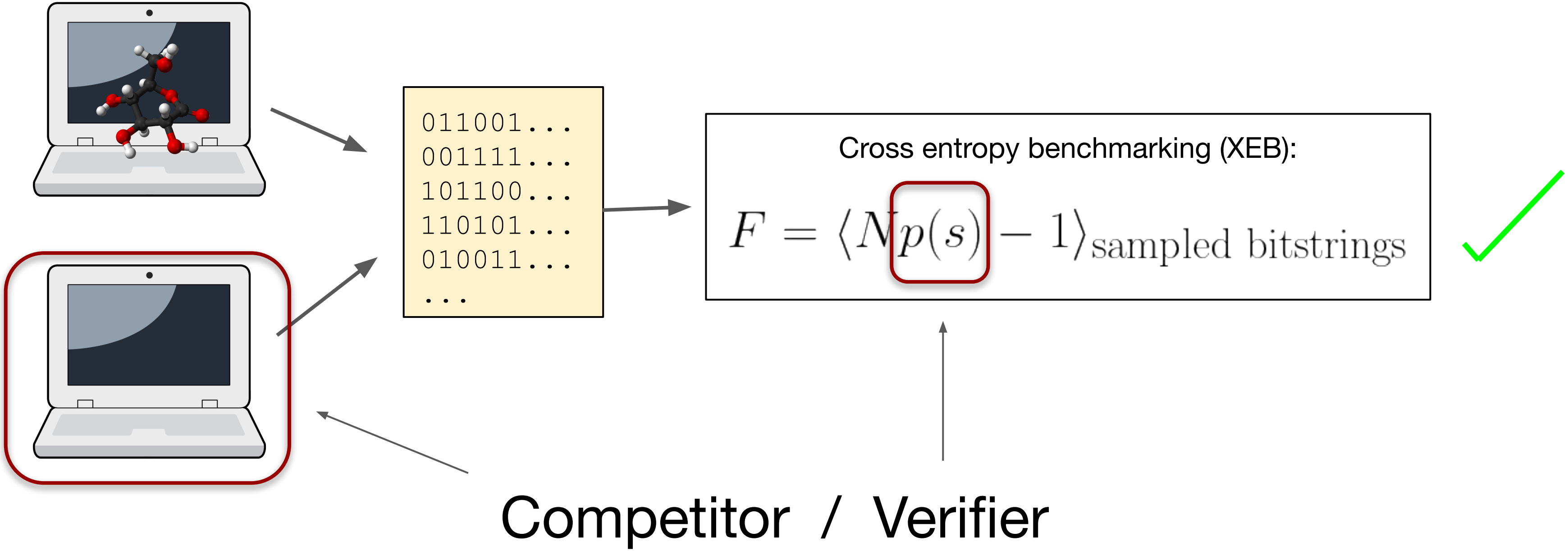
Coupler

Forward-compatible
with surface code

# PART 2

## Simulations

# The role of classical simulations



011001...
001111...
101100...
110101...
010011...
...

Cross entropy benchmarking (XEB):

$$F = \langle N \boxed{p(s)} - 1 \rangle_{\text{sampled bitstrings}}$$

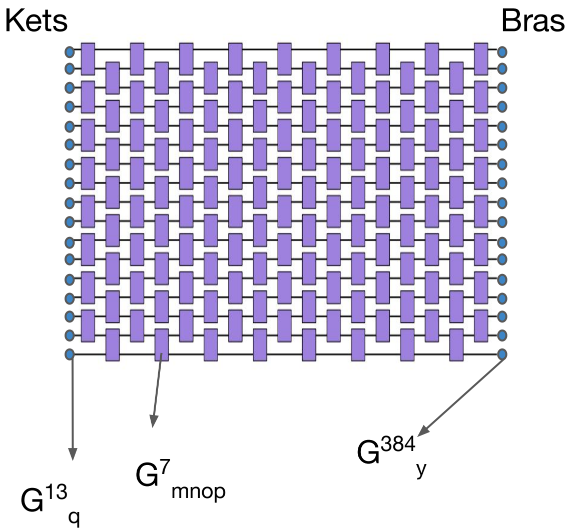
✓

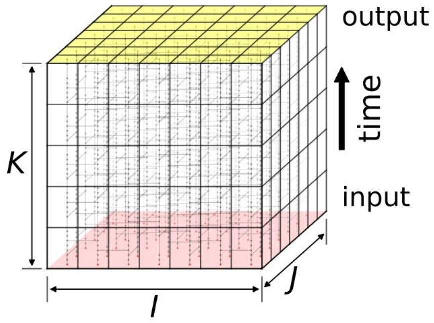
Competitor / Verifier

In both cases we have to compute **p**’s

# Tensor networks: a framework for (most) simulators

# Quantum circuits and tensor networks





$$= \sum_{abcde\ldots} G^1_{ab} G^2_{bcde} G^3_{cefg} \cdots$$

$$= \sum_{f} \sum_{g} \sum_{b} \sum_{a} \sum_{c} \cdots G^1_{ab} G^2_{bcde} G^3_{cefg} \cdots$$

$$= \sum_{f} \sum_{g} \left[ \sum_{b} \sum_{a} \sum_{c} \cdots G^1_{ab} G^2_{bcde} G^3_{cefg} \cdots \right]$$

Finding best ordering is NP-hard:
treewidth of line-graph of the TN

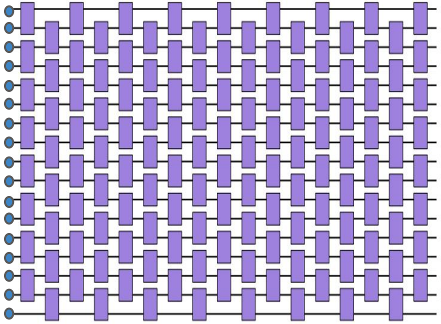
*Markov & Shi, 2008*

Label sub tensor networks
(parallel processes)

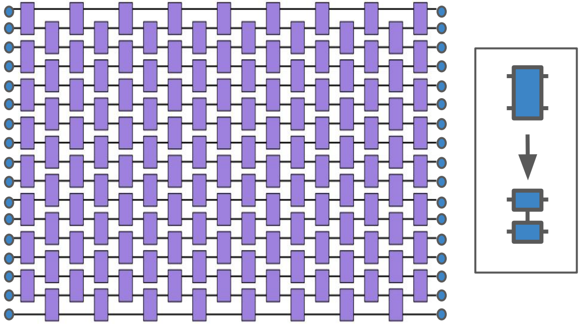
Independent summations
(also less complex)

*Chen et al., 2018*

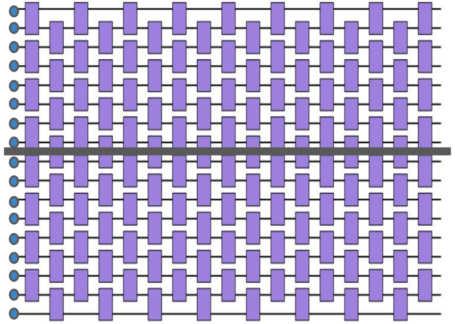
# Types of simulators (competitors / verifiers)



Schroedinger



Feynman



Shroedinger-Feynman

### Typical time evolution

**Advantages:**

- Best for deep circuits
- Get all amplitudes at once
- No need to find contraction ordering

**Disadvantages:**

- Need to store the full wave function
- Time and memory exponential in #qubits

### Contract in exotic ways...

**Advantages:**

- Best for shallow circuits
- Lower memory
- Cost proportional to F

**Disadvantages:**

- One amplitude at a time
- Time and memory “exponential” in #qubits *and* depth
- Finding best contraction is NP-hard

### Hybrid (many small evolutions)

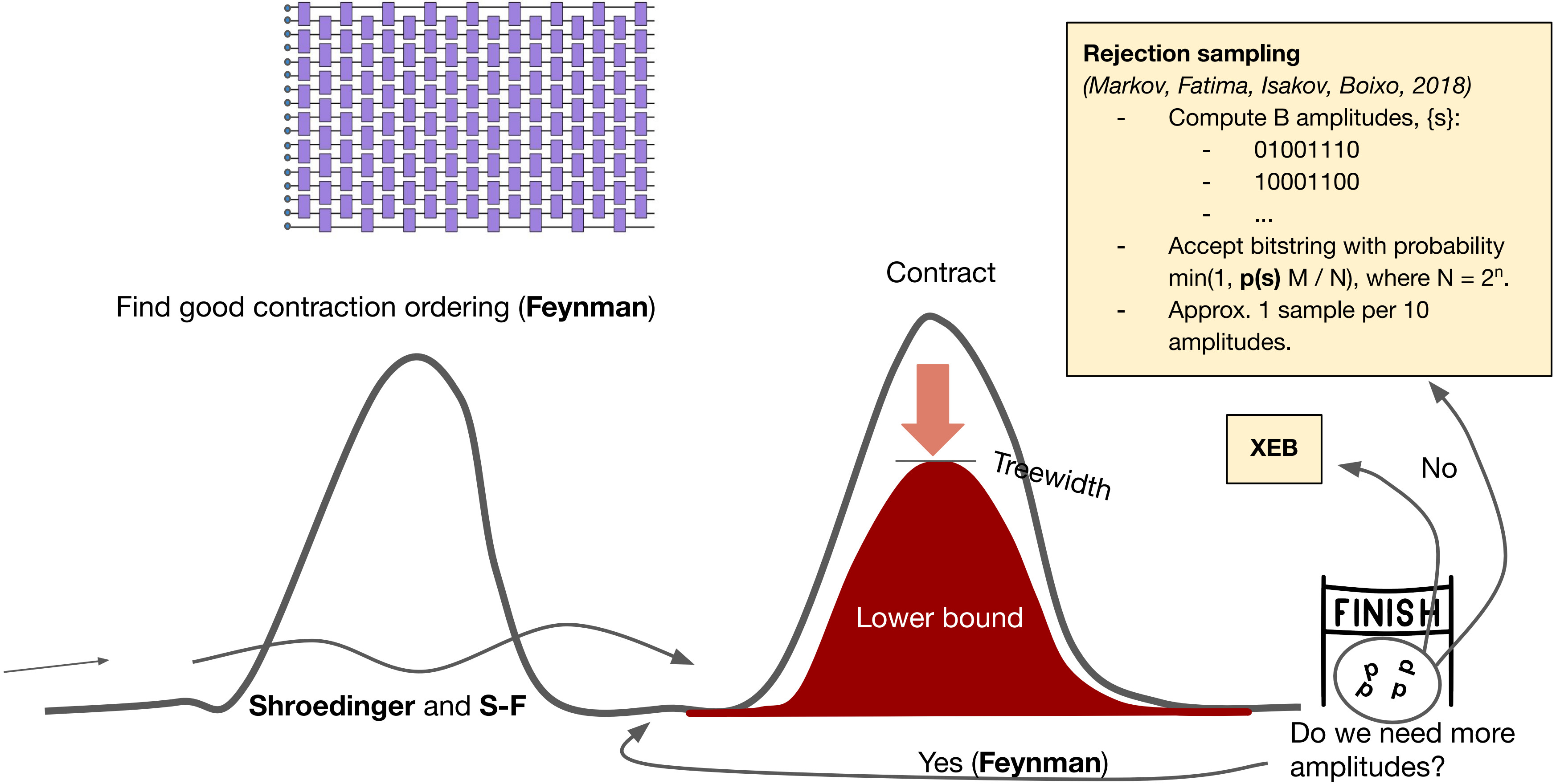
**Advantages:**

- Best for moderate depth
- Get all amplitudes at once
- Lower memory
- Cost proportional to F

**Disadvantages:**

- Time and memory “exponential” in #qubits *and* depth

# The effort of simulating quantum circuits



**Rejection sampling**
*(Markov, Fatima, Isakov, Boixo, 2018)*

- Compute B amplitudes, {s}:
  - 01001110
  - 10001100
  - ...
- Accept bitstring with probability min(1, **p(s)** M / N), where N = $2^n$.
- Approx. 1 sample per 10 amplitudes.

Find good contraction ordering (**Feynman**)

Contract

Treewidth

Lower bound

**Shroedinger** and **S-F**

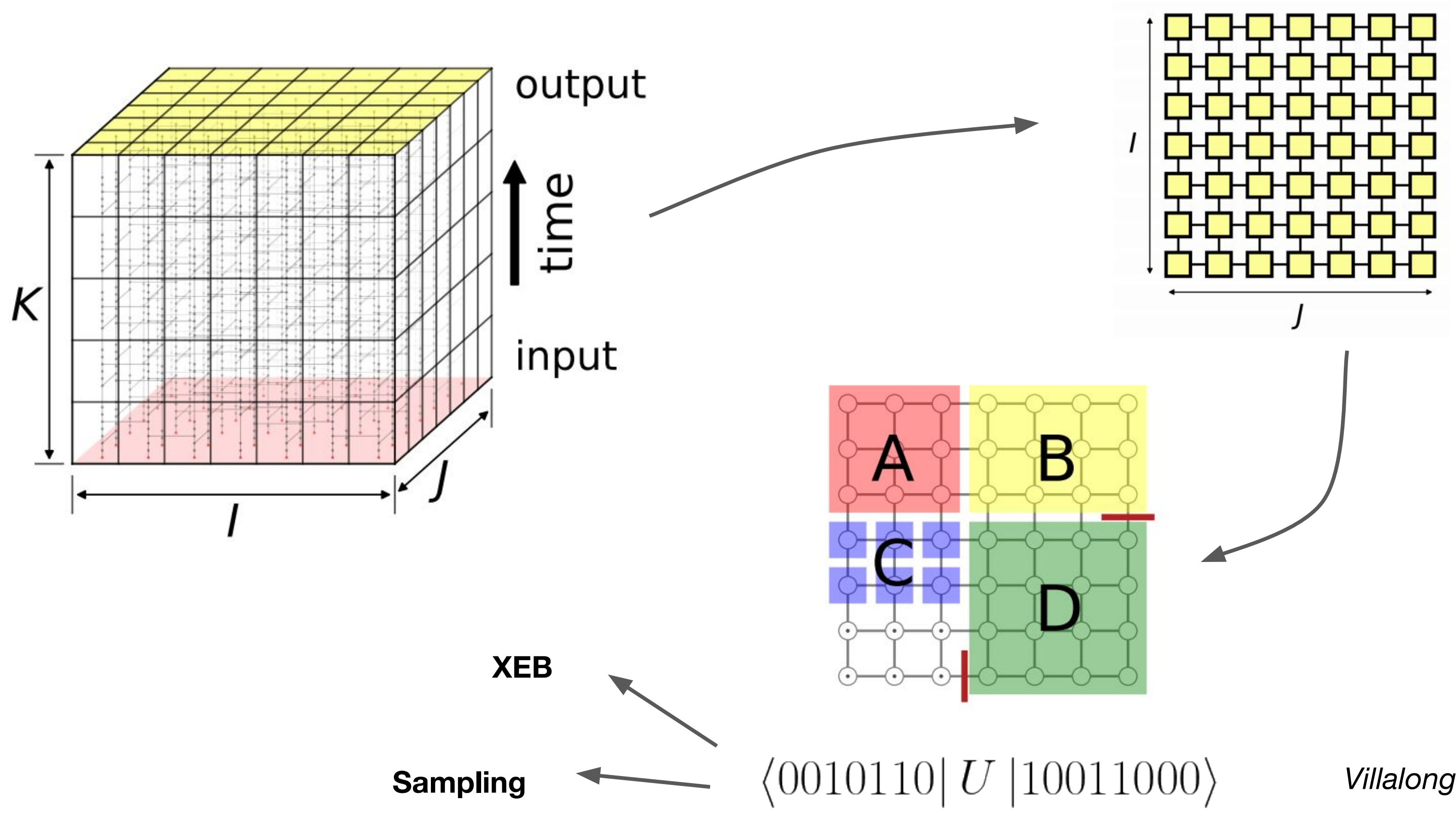
**XEB**

No

Do we need more amplitudes?

Yes (**Feynman**)

# Contraction



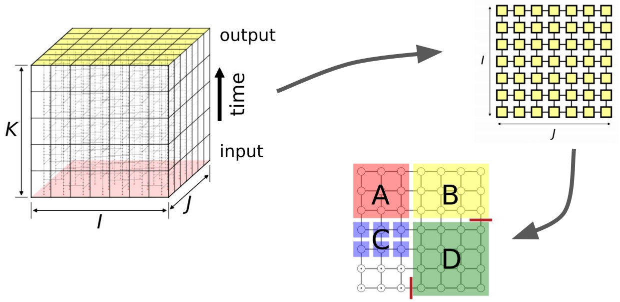
XEB

Sampling

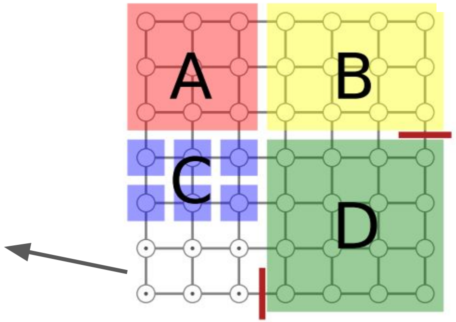
$$\langle 0010110 | \, U \, | 10011000 \rangle$$

*Villalonga et al., 2018*

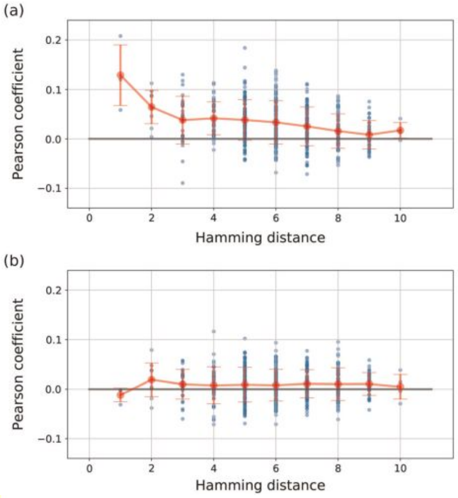
# Towards more efficient sampling



$$\langle 00101100 |\, U \,| 00000000 \rangle$$



**Batch** of amplitudes



*Villalonga et al., 2018*

On Summit (GPUs) we run up to 121 qubit shallow QCs (70-90% efficiency)

*Villalonga et al., 2019*

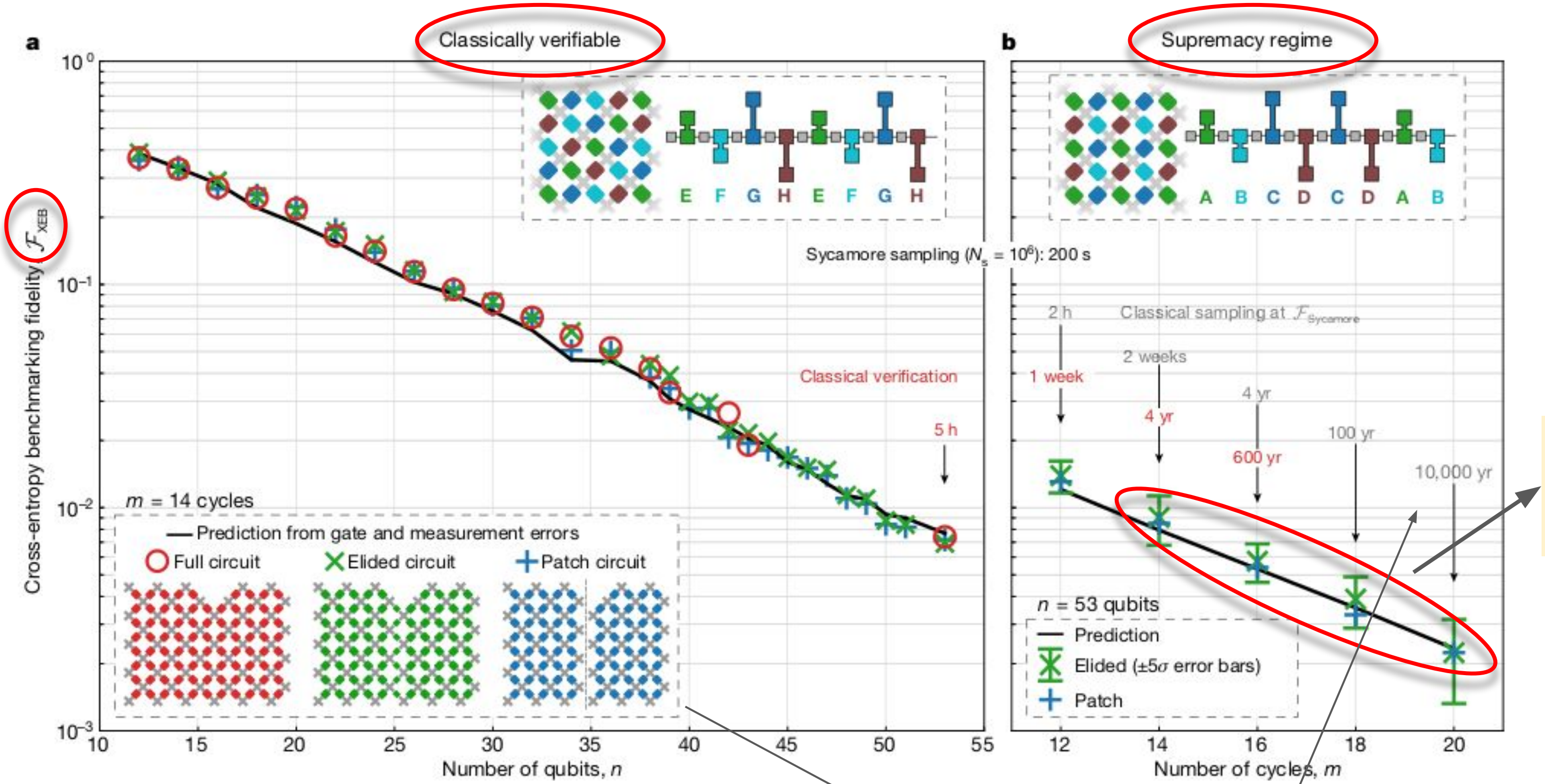
**Rejection sampling**
*(Markov, Fatima, Isakov, Boixo, 2018)*

- Compute B amplitudes, {s}:
  - 01001110
  - 10001100
  - ...
- Accept bitstring with probability min(1, **p(s)** M / N), where N = $2^n$.
- Approx. **1 sample** per ~~10 amplitudes~~ **1 batch**.

**Low fidelity sampling**

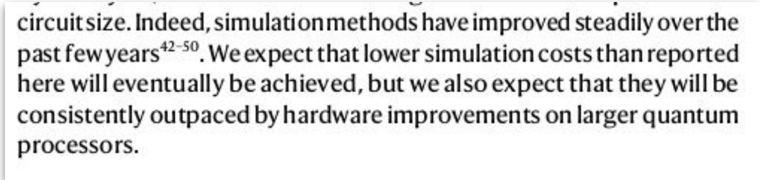
Sampling correctly a fraction **F** of the time yields fidelity **F** (linear speedup)

Reminiscent of:
*Markov, 2018*



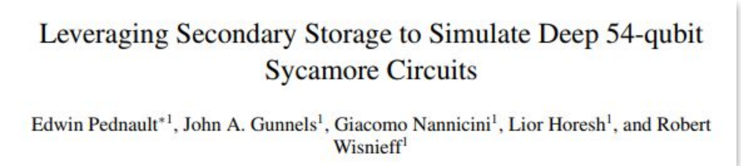
Not verifiable,
*but archived*

**Schroedinger-Feynman**



circuit size. Indeed, simulation methods have improved steadily over the past few years[42–50]. 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.

# Post-experiment follow-up



## Leveraging Secondary Storage to Simulate Deep 54-qubit Sycamore Circuits

Edwin Pednault*[1], John A. Gunnels[1], Giacomo Nannicini[1], Lior Horesh[1], and Robert Wisnieff[1]

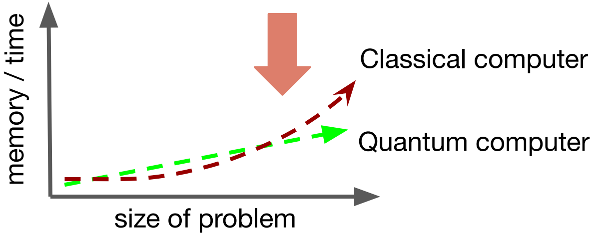
Use the many PB disk!

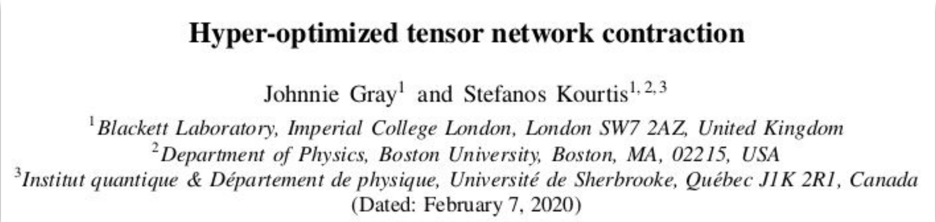
**How hard are simulations?**

Originally thousands of years (hundreds)

Weeks? days? vs. 200s

~10 MW vs. ~10 KW



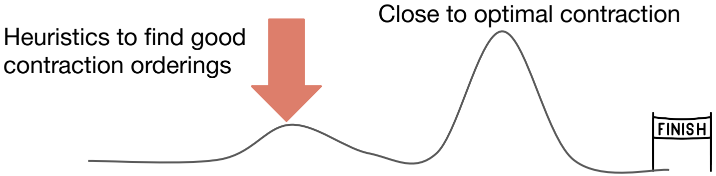


## Hyper-optimized tensor network contraction

Johnnie Gray[1] and Stefanos Kourtis[1, 2, 3]

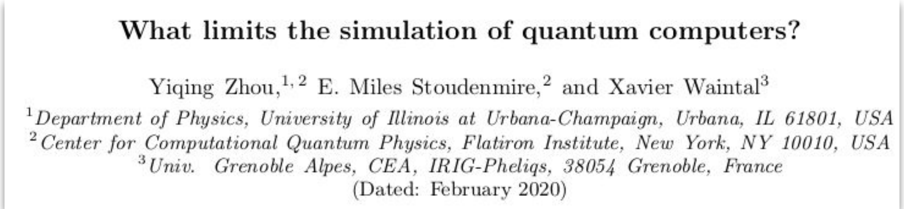
[1] *Blackett Laboratory, Imperial College London, London SW7 2AZ, United Kingdom*
[2] *Department of Physics, Boston University, Boston, MA, 02215, USA*
[3] *Institut quantique & Département de physique, Université de Sherbrooke, Québec J1K 2R1, Canada*

(Dated: February 7, 2020)



Heuristics to find good contraction orderings

Close to optimal contraction

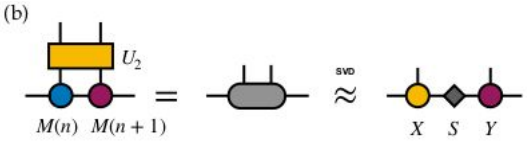


## What limits the simulation of quantum computers?

Yiqing Zhou,[1, 2] E. Miles Stoudenmire,[2] and Xavier Waintal[3]

[1] *Department of Physics, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA*
[2] *Center for Computational Quantum Physics, Flatiron Institute, New York, NY 10010, USA*
[3] *Univ. Grenoble Alpes, CEA, IRIG-Pheliqs, 38054 Grenoble, France*

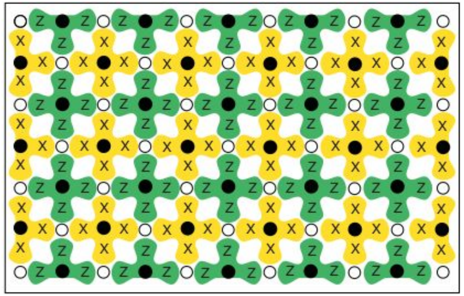
(Dated: February 2020)



For **F = 1 - ε**, cost is linear in 1D

# What’s next?

*Fowler et al., 2012*

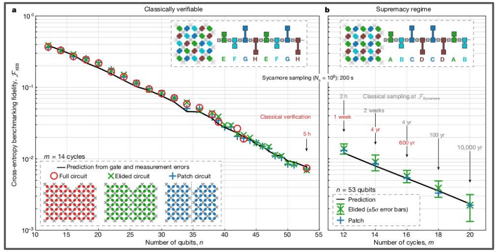


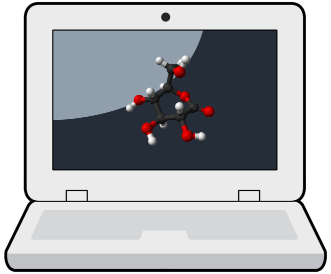
**Long term:** error correction

---

**Near term:** NISQ era just got unlocked!

Useful supremacy: optimization, quantum chemistry, certified random number generation, ...





Hello
World!