Title: Quantum Computational Supremacy and Its Applications

Speakers: Scott Aaronson

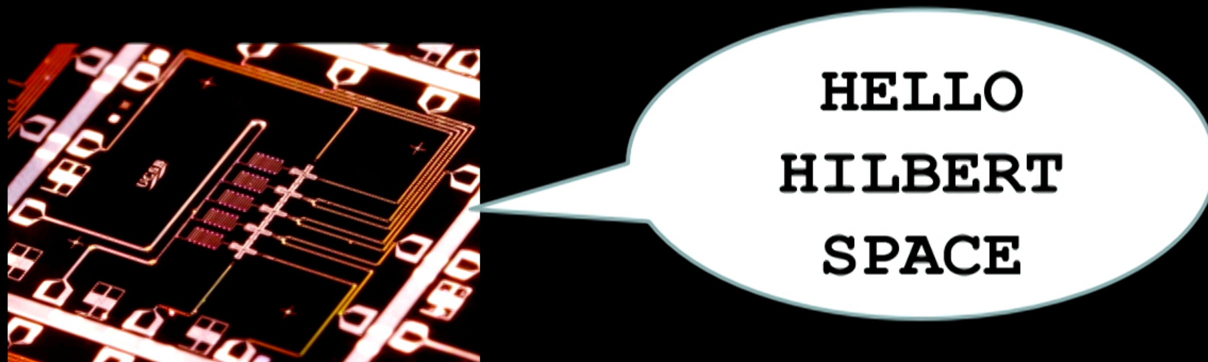
Series: Colloquium

Date: January 29, 2020 - 2:00 PM

URL: http://pirsa.org/20010094

Abstract: Last fall, a team at Google announced the first-ever demonstration of "quantum computational supremacy"---that is, a clear quantum speedup over a classical computer for some task---using a 53-qubit programmable superconducting chip called Sycamore. In addition to engineering, Google's accomplishment built on a decade of research in quantum computing theory. This talk will discuss questions like: what exactly was the contrived computational problem that Google solved? How does one verify the outputs using a classical computer? And how confident are we that the problem really is classically hard---especially in light of subsequent counterclaims by IBM? I'll end with a proposed application for Google's experiment---namely, the generation of certified random bits, for use (for example) in proof-of-stake cryptocurrencies---that I've been developing and that Google is now working to demonstrate.

Quantum Computational Supremacy and Its Applications



Scott Aaronson (University of Texas at Austin)

Perimeter Institute, January 29, 2020

Quantum Computational Supremacy and Its Applications

Scott Aaronson (University of Texas at Austin)

Perimeter Institute, January 29, 2020

Quantum technologies + Add to myFT

Google claims to have reached quantum supremacy

MOTHERBOARD
TECH BY VICE

OK, WTF Is Google's 'Quantum Supremacy'?

Google may have just ushered in an era 'quantum supremacy'

'The first computation that can only be performed on a quantum processor'

By Jon Porter | @JonPorty | Sep 23, 2019, 7:06am EDT

SHARE

Google has reached quantum supremacy – here's what it should do next

TECHNOLOGY | ANALYSIS 26 September 2019

By Chelsea Whyte

Google's 'Quantum Supremacy' Isn't the End of Encryption

Google said its quantum computer outperformed conventional models. But it will : anything practical.

NEWS QUANTUM PHYSICS

Rumors hint that Google has accomplished quantum supremacy

Reports suggest a quantum computer has surpassed standard computers on a specific type of calculation


MOTHERBOARD
TECH BY VICE
OK, WTF Is Google's 'Quantum Supremacy'?
Quantum technologies
+ Add to myFT
Google claims to have reached quantum
supren
Google m
'quantum
'The first computati
By Jon Porter | @JonPorty |
SHARE
Google's 'Qu
Encryption
Google said its quantum co
anything practical.
Shtetl-Optimized
The Blog of Scott Aaronson
If you take just one piece of information from this blog:
Quantum computers would not solve hard search problems
instantaneously by simply trying all the possible solutions at once.
PSPACE
PostBQP
NP
BQP
P
« Blurry but clear enough
From quantum supremacy to classical fallacy »
Scott's Supreme Quantum Supremacy FAQ!
You've seen the stories—in the Financial Times, Technology Review, CNET, Facebook, Reddit, Twitter, or elsewhere—saying that a group at Google has now achieved quantum computational supremacy with a 53-qubit superconducting device. While these stories are easy to find, I'm not going to link to them here, for the simple reason that none of them were supposed to exist yet.
As the world now knows, Google is indeed preparing a big announcement about quantum supremacy, to coincide with the publication of its research paper in a high-profile journal (which journal? you can probably narrow it down to two). This
ıtum
t it should
oogle
has accomplished quantum
supremacy
Reports suggest a quantum computer has surpassed standard computers on a specific type of calculation

What Is Quantum Computational Supremacy?

Preskill 2012

The use of a quantum computer to solve **some** well-defined problem much faster than any available classical computer running any known algorithm

What Is Quantum Computational Supremacy?

Preskill 2012

The use of a quantum computer to solve **some** well-defined problem much faster than any available classical computer running any known algorithm

- Problem doesn't need to be useful, but **does** need a hardware-independent definition (no "simulate yourself")

What Is Quantum Computational Supremacy?

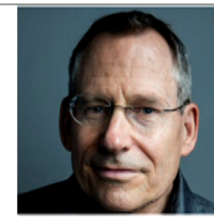
Preskill 2012

The use of a quantum computer to solve **some** well-defined problem much faster than any available classical computer running any known algorithm

- Problem doesn't need to be useful, but **does** need a hardware-independent definition (no "simulate yourself")

- Need: an actual observed speedup, **and** a theoretical asymptotic speedup, **and** a causal connection between the two

What Is Quantum Computational Supremacy?



Preskill 2012

The use of a quantum computer to solve **some** well-
defi
clas

"#1 Application of a Quantum Computer: Refute those who said it was impossible!"

- d

-

theoretical asymptotic speedup, **and** a causal connection between the two

The Google Team's Results

The Google Team's Results

Built "Sycamore," a 53-qubit superconducting chip with controllable 2D nearest-neighbor couplings

The Google Team's Results

Built "Sycamore," a 53-qubit superconducting chip with controllable 2D nearest-neighbor couplings

Used it to output samples from some distribution D over 53-bit strings **(or rather, from ~0.998U+0.002D, where U is the uniform distribution)**

The Google Team's Results

Built "Sycamore," a 53-qubit superconducting chip with controllable 2D nearest-neighbor couplings

Used it to output samples from some distribution D over 53-bit strings **(or rather, from ~0.998U+0.002D, where U is the uniform distribution)**

Took 5 million samples in ~3 minutes (~40 microseconds per sample); did statistics on them to extract a signal corresponding to D

Argued that the best current classical algorithms, running on 1M cores, would've taken ~10,000 years to do the same—so, a quantum speedup by a factor of ~2 billion, or ~10^{15} in the "number of gates"

How to Achieve Quantum Supremacy

Shor's factoring algorithm, Grover's search algorithm, even simulating quantum chemistry: Alas, too expensive for now. May require error-correction, incurring a huge overhead—but at any rate, more and better qubits than feasible in the near future.

How to Achieve Quantum Supremacy

Shor's factoring algorithm, Grover's search algorithm, even simulating quantum chemistry: Alas, too expensive for now. May require error-correction, incurring a huge overhead—but at any rate, more and better qubits than feasible in the near future.

Idea (Terhal-DiVincenzo 2004, A.-Arkhipov 2011, Bremner-Jozsa-Shepherd 2011): Sampling problems. Instead of a single right answer, just ask for samples from a target probability distribution

How to Achieve Quantum Supremacy

Shor's factoring algorithm, Grover's search algorithm, even simulating quantum chemistry: Alas, too expensive for now. May require error-correction, incurring a huge overhead—but at any rate, more and better qubits than feasible in the near future.

Idea (Terhal-DiVincenzo 2004, A.-Arkhipov 2011, Bremner-Jozsa-Shepherd 2011): Sampling problems. Instead of a single right answer, just ask for samples from a target probability distribution

Advantages: experimental feasibility, high confidence in classical hardness

How to Achieve Quantum Supremacy

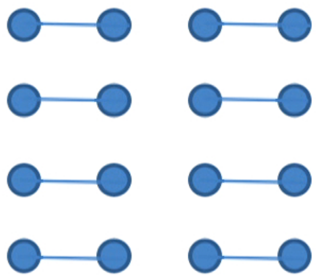
Shor's factoring algorithm, Grover's search algorithm, even simulating quantum chemistry: Alas, too expensive for now. May require error-correction, incurring a huge overhead—but at any rate, more and better qubits than feasible in the near future.

Idea (Terhal-DiVincenzo 2004, A.-Arkhipov 2011, Bremner-Jozsa-Shepherd 2011): Sampling problems. Instead of a single right answer, just ask for samples from a target probability distribution

Advantages: experimental feasibility, high confidence in classical hardness

Disadvantages: useless? how to check results?

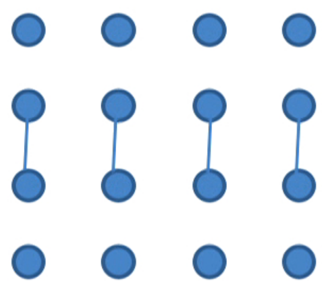
The Random Circuit Proposal



Challenge the QC by sending it a randomly generated quantum circuit C on n qubits

Ask the QC to send back (quickly!) samples $s_1,\ldots,s_k$ from D_C, the distribution over n-bit strings obtained by applying C to 0…0

The Random Circuit Proposal



Challenge the QC by sending it a randomly generated quantum circuit C on n qubits

Ask the QC to send back (quickly!) samples $s_1,\ldots,s_k$ from D_C, the distribution over n-bit strings obtained by applying C to 0…0

Then, using classical brute force, check if

$$\sum_{i=1}^{k} \left| \left\langle 0 \cdots 0 \middle| C \middle| s_i \right\rangle \right|^2 \geq \frac{bk}{2^n}$$

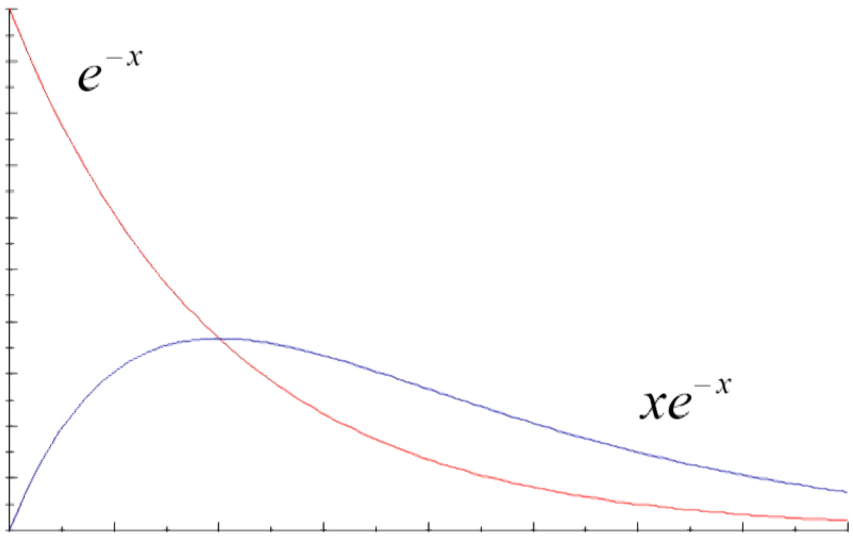
for some constant $b\in(1,2)$

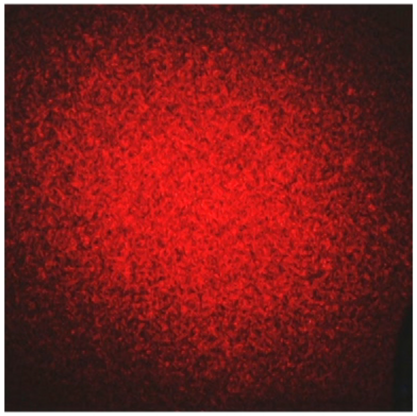
Google's "Linear Cross-Entropy Benchmark"

What's Going On

All 2^n possible output strings are exponentially unlikely—but some are unlikelier than others!

Picking uniformly at random will get samples of average probability $1/2^n$, but an ideal QC will get $2/2^n$





"Speckle"

IBM's Response

Using Summit, the largest supercomputer currently on earth—which fills 2 basketball courts and has 250 petabytes of hard disk—it should be possible to simulate Google's 3-minute calculation in ~2.5 days, rather than the 10,000 years Google estimated

Is There A 2^n Barrier?

Classical Simulation Algorithm	Time	Memory
Schrödinger	$\sim 2^n$ (n=#qubits)	$\sim 2^n$
Feynman	$\sim 2^m$ (m=#gates)	Linear
Schrödinger-Feynman (A.-Chen 2017)	$\sim d^n$ (d=depth)	Linear

Is There A 2^n Barrier?

Classical Simulation Algorithm	Time	Memory
Schrödinger	$\sim 2^n$ (n=#qubits)	$\sim 2^n$
Feynman	$\sim 2^m$ (m=#gates)	Linear
Schrödinger-Feynman (A.-Chen 2017)	$\sim d^n$ (d=depth)	Linear

Theorem (A.-Chen 2017, A.-Gunn 2019): If there's a classical algorithm to spoof Linear XEB in $\ll 2^n$ time, then there's *also* a fast classical algorithm that estimates a *specific* output probability like $|\langle 0^n|C|0^n\rangle|^2$, with *slightly* better variance than always guessing 2^{-n}

Is There A 2^n Barrier?

Classical Simulation Algorithm	Time	Memory
Schrödinger	$\sim 2^n$ (n=#qubits)	$\sim 2^n$
Feynman	$\sim 2^m$ (m=#gates)	Linear
Schrödinger-Feynman (A.-Chen 2017)	$\sim d^n$ (d=depth)	Linear

Theorem (A.-Chen 2017, A.-Gunn 2019): If there's a classical algorithm to spoof Linear XEB in $\ll 2^n$ time, then there's *also* a fast classical algorithm that estimates a *specific* output probability like $|\langle 0^n|C|0^n\rangle|^2$, with *slightly* better variance than always guessing 2^{-n}

Proof Idea: First hide which output z you care about. Then run the spoofing algorithm and see if it outputs z

“But even if these sampling-based supremacy experiments work, they’ll just produce mostly-random bits, which is *obviously* useless…”

“But even if these sampling-based supremacy experiments work, they’ll just produce mostly-random bits, which is *obviously* useless…”

11010000110100111101101100110011000101001001

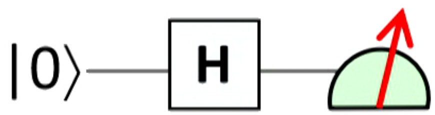
New Idea (A. 2018): Randomness from Quantum Supremacy Experiments

New Idea (A. 2018): Randomness from Quantum Supremacy Experiments

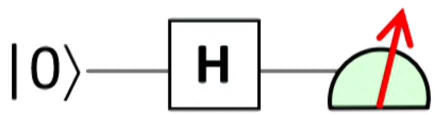
Key Insight: If a QC passes the Linear XEB test for quantum supremacy, it's proved more than just its quantumness. Under plausible assumptions, it's also proved it must have generated the samples **randomly**!

(At least somewhat)

Obviously, there are **much** easier ways to generate random bits!

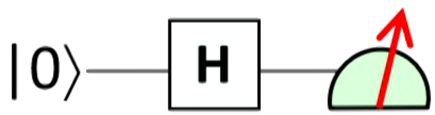


Obviously, there are **much** easier ways to generate random bits!



These are fine if you own your own random number generator. But what if you need to download random bits off the Internet and **trust** that they're random? This is exactly the situation with, e.g., **proof-of-stake cryptocurrencies** and other crypto protocols

Obviously, there are **much** easier ways to generate random bits!



These are fine if you own your own random number generator. But what if you need to download random bits off the Internet and **trust** that they're random? This is exactly the situation with, e.g., **proof-of-stake cryptocurrencies** and other crypto protocols

Google and NIST are currently working to demonstrate a public randomness beacon based on my protocol

Where To Go Next?

Replicate Google's result in other hardware platforms, and with higher fidelity, more qubits, less specialized gate set, independent challenges and verification…

Design new classical simulation algorithms! Especially for low-depth circuits (cf. Napp, La Placa, Dalzell, Brandao, Harrow)

Better complexity-theoretic evidence for hardness

Near-term quantum supremacy with efficient classical verification?

Generating more and more certified random bits by running the same circuit C over and over?

Conclusions

Google has apparently achieved quantum computational supremacy, using Random Circuit Sampling on 53 qubits—building on a lot of "useless complexity theory" we did over the past decade!

Even if true, this leaves the huge challenges of **scalability** and **fault-tolerance**. But it already refutes those who said quantum speedups are impossible

Conclusions

Google has apparently achieved quantum computational supremacy, using Random Circuit Sampling on 53 qubits—building on a lot of "useless complexity theory" we did over the past decade!

Even if true, this leaves the huge challenges of **scalability** and **fault-tolerance**. But it already refutes those who said quantum speedups are impossible

It was thought obvious for years that sampling-based supremacy experiments had no applications. My certified randomness protocol may change that—though challenges remain in making it practical