TAURUS

Post-Quantum Crypto is Coming!

JP Aumasson

P2P Paris – 2022-05-01

Background

Co-founder & chief security officer of Taurus SA

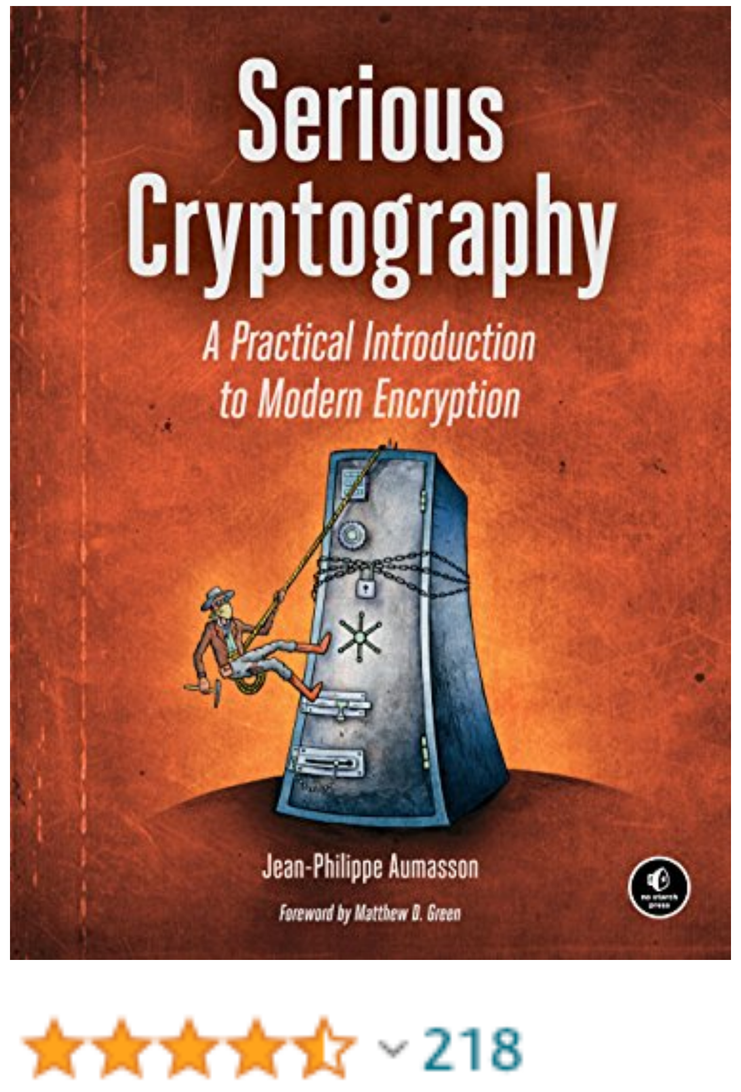
- Swiss firm founded in 2018, team of 40+
- Crypto custody technology and infrastructure , FINMA licensed
- Taurus used by all types of banks and financial institutions

https://taurushq.com https://t-dx.com

Expert in cryptography and security

- 15 years in crypto and security, EPFL PhD
- Designed algorithms used in Linux, Bitcoin, etc.
- Author of reference books in the field

https://aumasson.jp https://twitter.com/veorq



Prerequisites

Fundamental Equations

Schrödinger equation:

$$i\hbar\frac{\partial\Psi}{\partial t} = H\Psi$$

Time independent Schrödinger equation:

$$H\psi = E\psi, \qquad \Psi = \psi e^{-iEt/\hbar}$$

Standard Hamiltonian:

$$H = -\frac{\hbar^2}{2m}\nabla^2 + V$$

Time dependence of an expectation value:

$$\frac{d\langle Q\rangle}{dt} = \frac{i}{\hbar}\langle[H,Q]\rangle + \left\langle\frac{\partial Q}{\partial t}\right\rangle$$

Generalized uncertainty principle:

$$\sigma_A\sigma_B \geq \left|\frac{1}{2i}\langle[A,B]\rangle\right|^2$$

Why Quantum Computers?

Simulating Physics with Computers

Richard P. Feynman

Department of Physics, California Institute of Technology, Pasadena, California 91107

Received May 7, 1981

Not to Break Crypto..

5. CAN QUANTUM SYSTEMS BE PROBABILISTICALLY SIMULATED BY A CLASSICAL COMPUTER?

Now the next question that I would like to bring up is, of course, the interesting one, i.e., Can a quantum system be probabilistically simulated by a classical (probabilistic, I'd assume) universal computer? In other words, a computer which will give the same probabilities as the quantum system does. If you take the computer to be the classical kind I've described so far, (not the quantum kind described in the last section) and there're no changes in any laws, and there's no hocus-pocus, the answer is certainly, No! This is called the hidden-variable problem: it is impossible to represent the results of quantum mechanics with a classical universal device. To learn a little bit about it, I say let us try to put the quantum equations in a form as close as

But (Initially) to Simulate Quantum Physics

4. QUANTUM COMPUTERS—UNIVERSAL QUANTUM SIMULATORS

The first branch, one you might call a side-remark, is, Can you do it with a new kind of computer—a quantum computer? (I'll come back to the other branch in a moment.) Now it turns out, as far as I can tell, that you can simulate this with a quantum system, with quantum computer elements. It's not a Turing machine, but a machine of a different kind. If we disregard the continuity of space and make it discrete, and so on, as an approximation (the same way as we allowed ourselves in the classical case), it does seem to

Qubits Instead of Bits

$\alpha\,|0\rangle + \beta\,|1\rangle$

Qubit state

Measure

0 with probability $|\alpha|^2$

1 with probability $|\beta|^2$

Qubit stays 0 or 1 forever

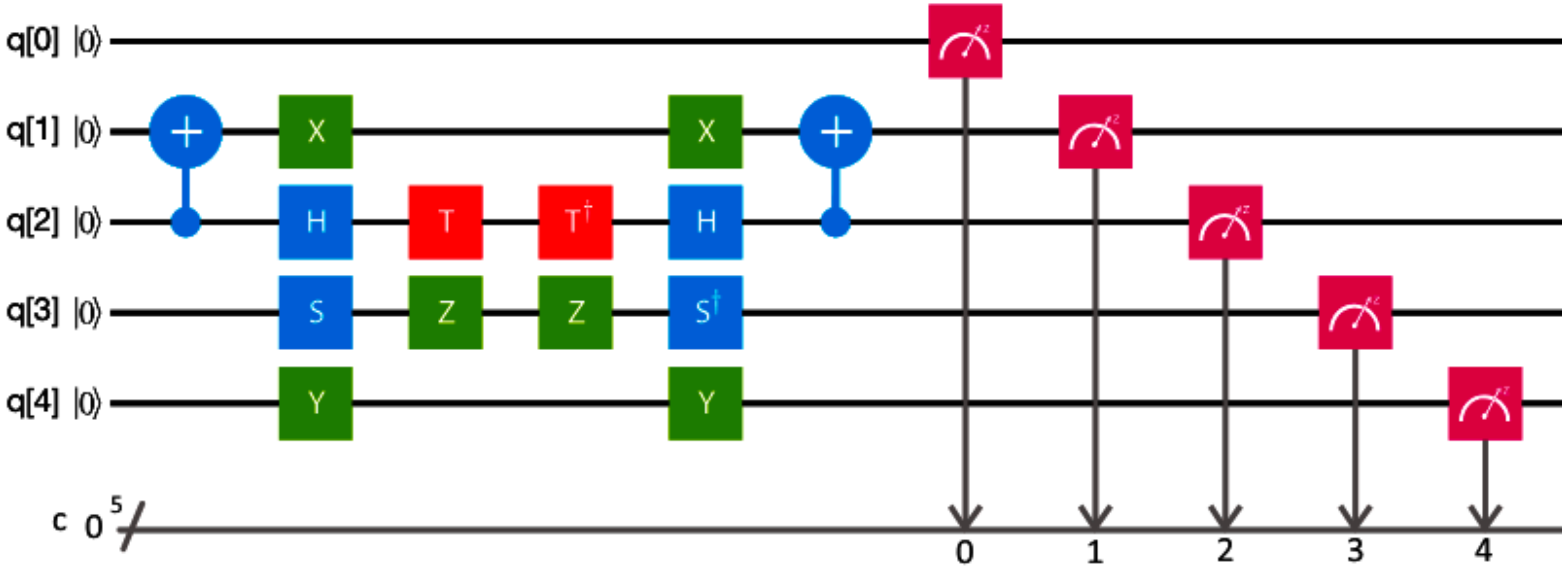
Generalizes to more than 2 states: qutrits, qubytes, etc.

α, β are complex, negative "probabilities" called **amplitudes**

Real randomness!

How Quantum Algorithms Work

Circuit of quantum gates, transforming a quantum state, ending with a measurement



Can be simulated with high-school linear algebra, but does no scale!

- **Quantum state** = vector of 2^N amplitudes for N qubits
- **Quantum gates** = matrix multiplications, with $O(2^{3N})$ complexity

Quantum Speedup

When quantum computers can solve a problem faster than classical computers

Most interesting: **Superpolynomial** quantum speedup (“exponential" boost)

List of problems on the Quantum Zoo: http://math.nist.gov/quantum/zoo/

Quantum Parallelism

Quantum computers “work" on all values simultaneously, via **superposition**

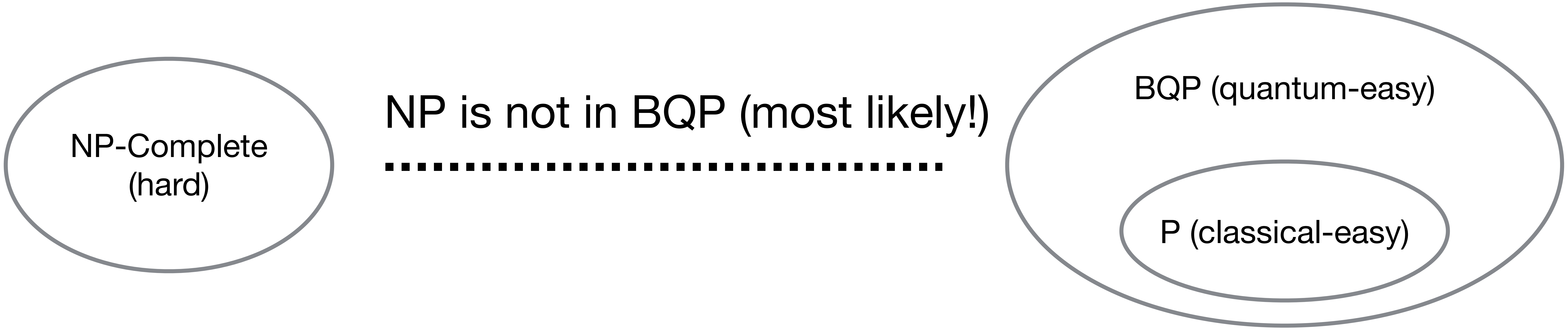
But they cannot *“try every answer in parallel and pick the best”*

You can only **observe one “value”** that results from the interference of all, through a projection from the Hilbert space (where qubits “live") to some basis

NP-complete Problems

- Solution hard to find, but easy to verify
- Constraint satisfaction problems (SAT, TSP, knapsacks, etc.)
- Sometimes used in crypto (lattice problems in post-quantum schemes)

Can't be solved faster with quantum computers!



BQP = bounded-error quantum polynomial time, what QC can solve efficiently

Quantum Supremacy?

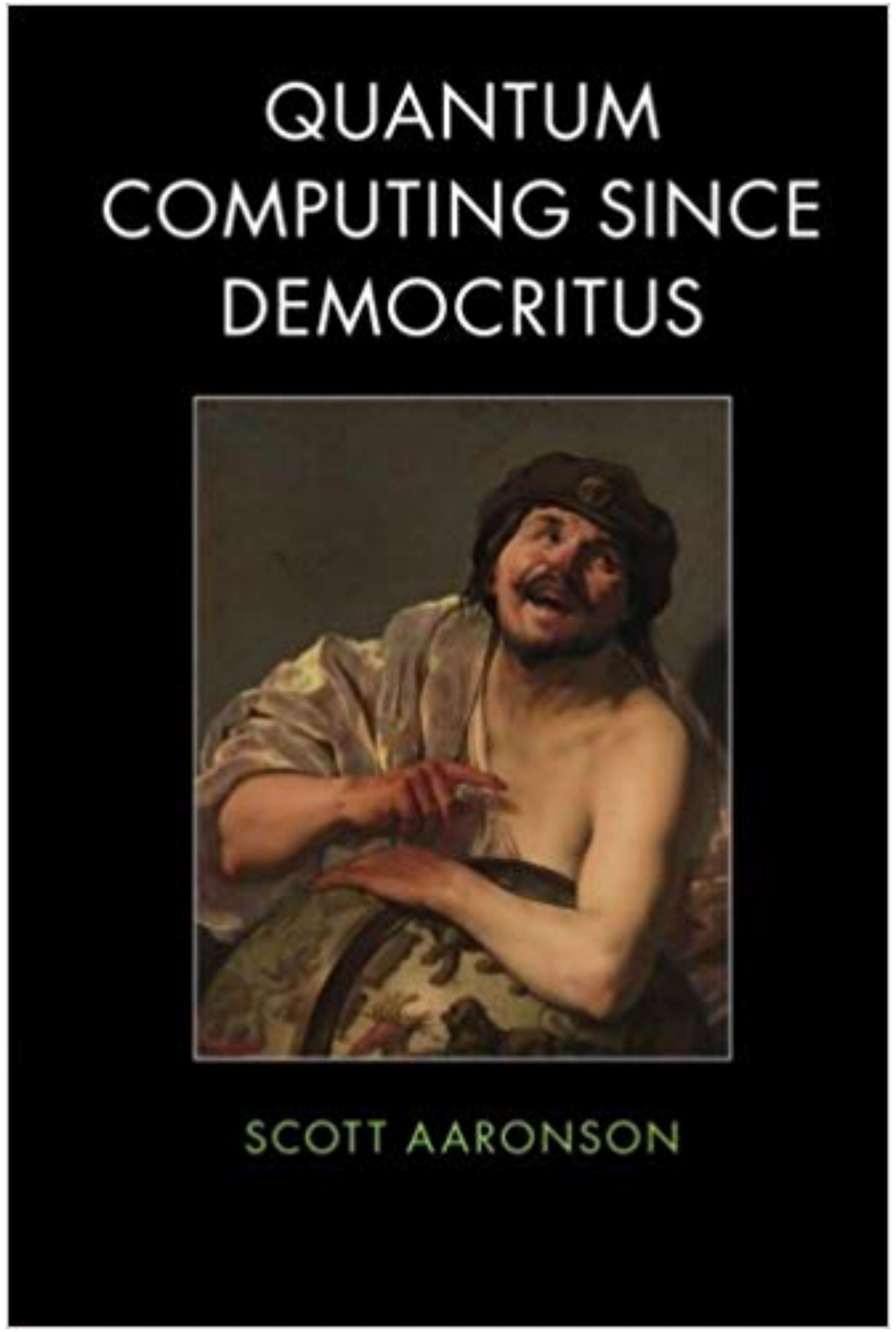
Google thinks it's close to "quantum supremacy." Here's what that really means.

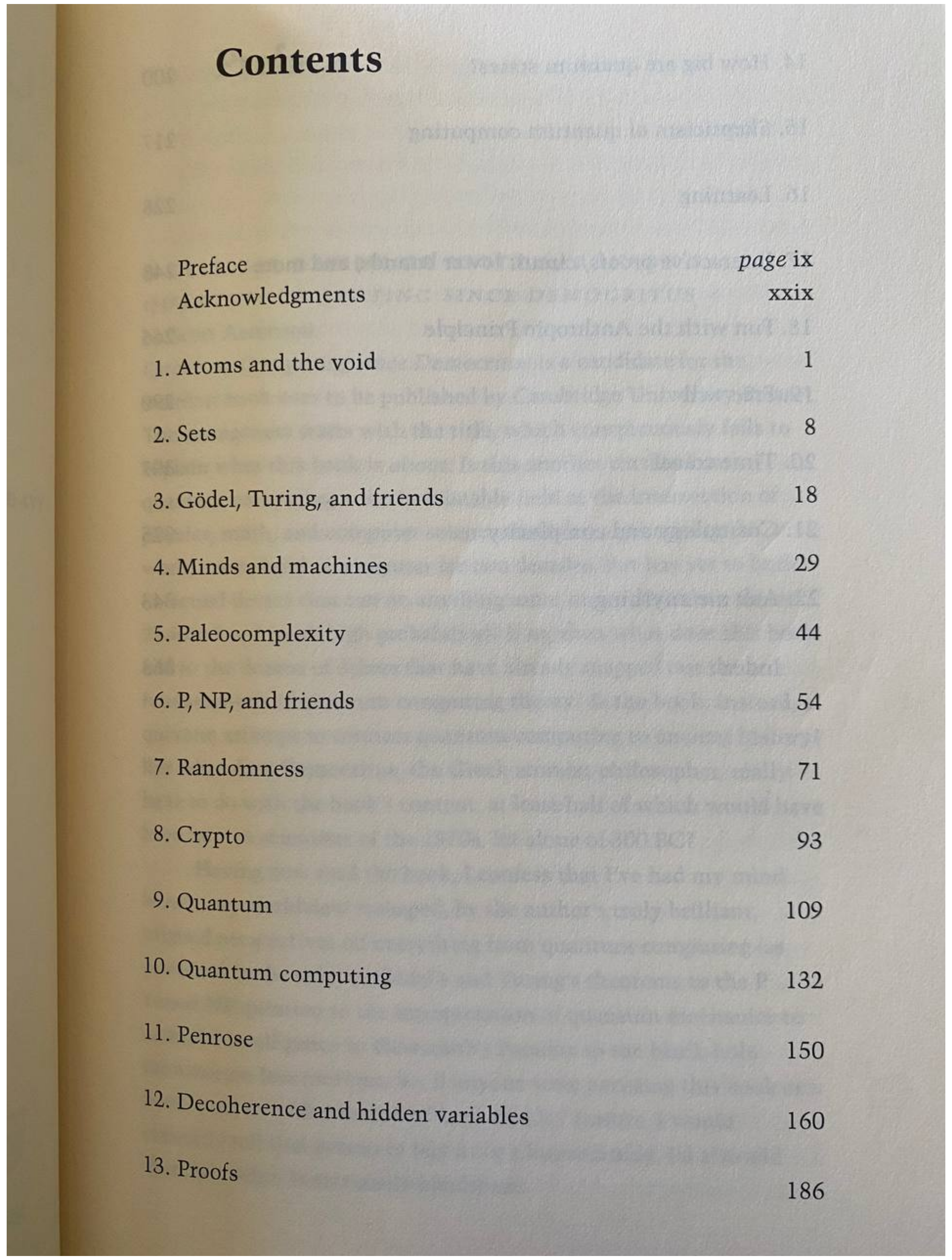
It's not the number of qubits; it's what you do with them that counts.

by Martin Giles and Will Knight   March 9, 2018

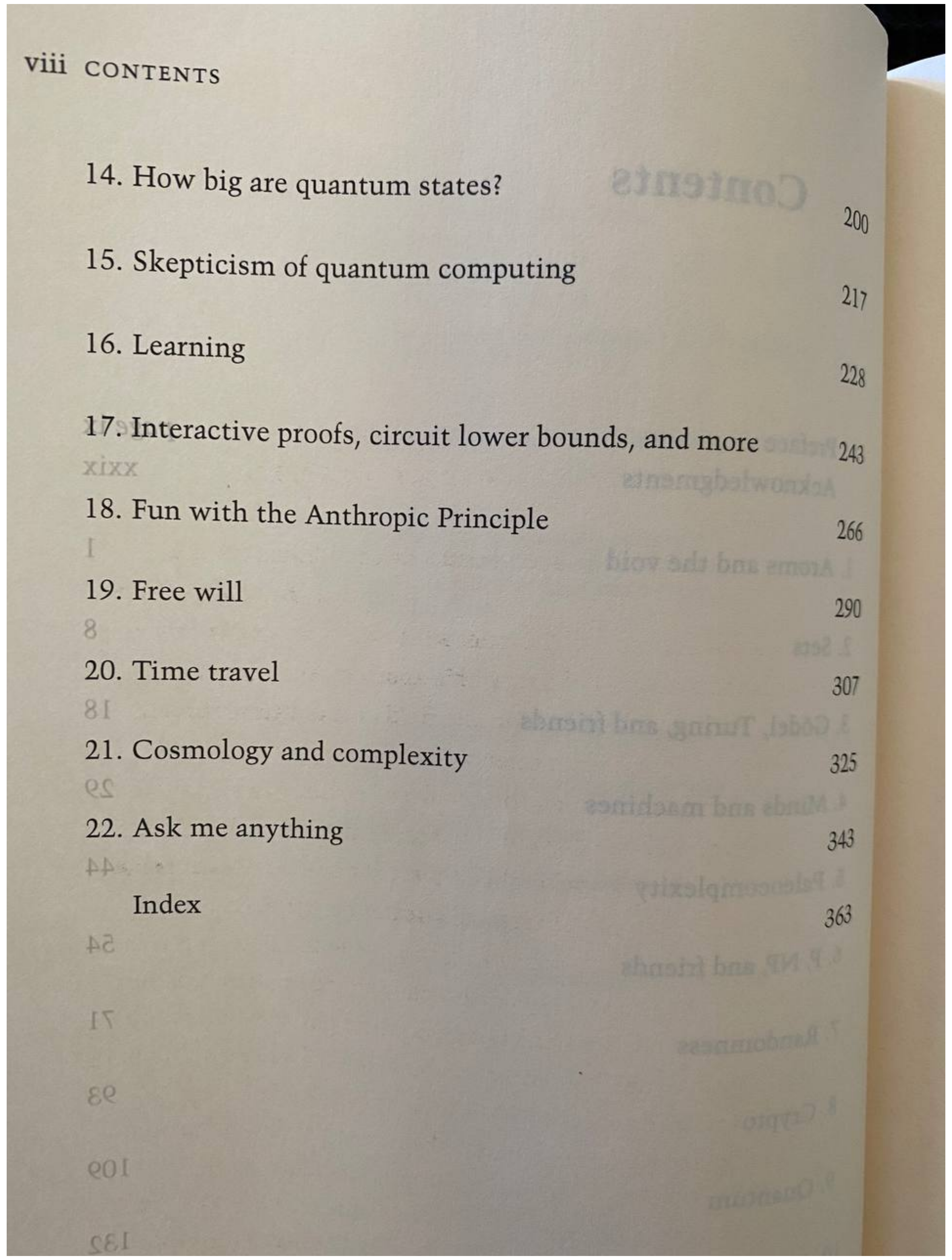
Seventy-two may not be a large number, but in quantum computing terms, it's massive. This week Google unveiled Bristlecone, a new quantum computing chip with 72 quantum bits, or qubits—the fundamental units of computation

Recommended Reading





Contents

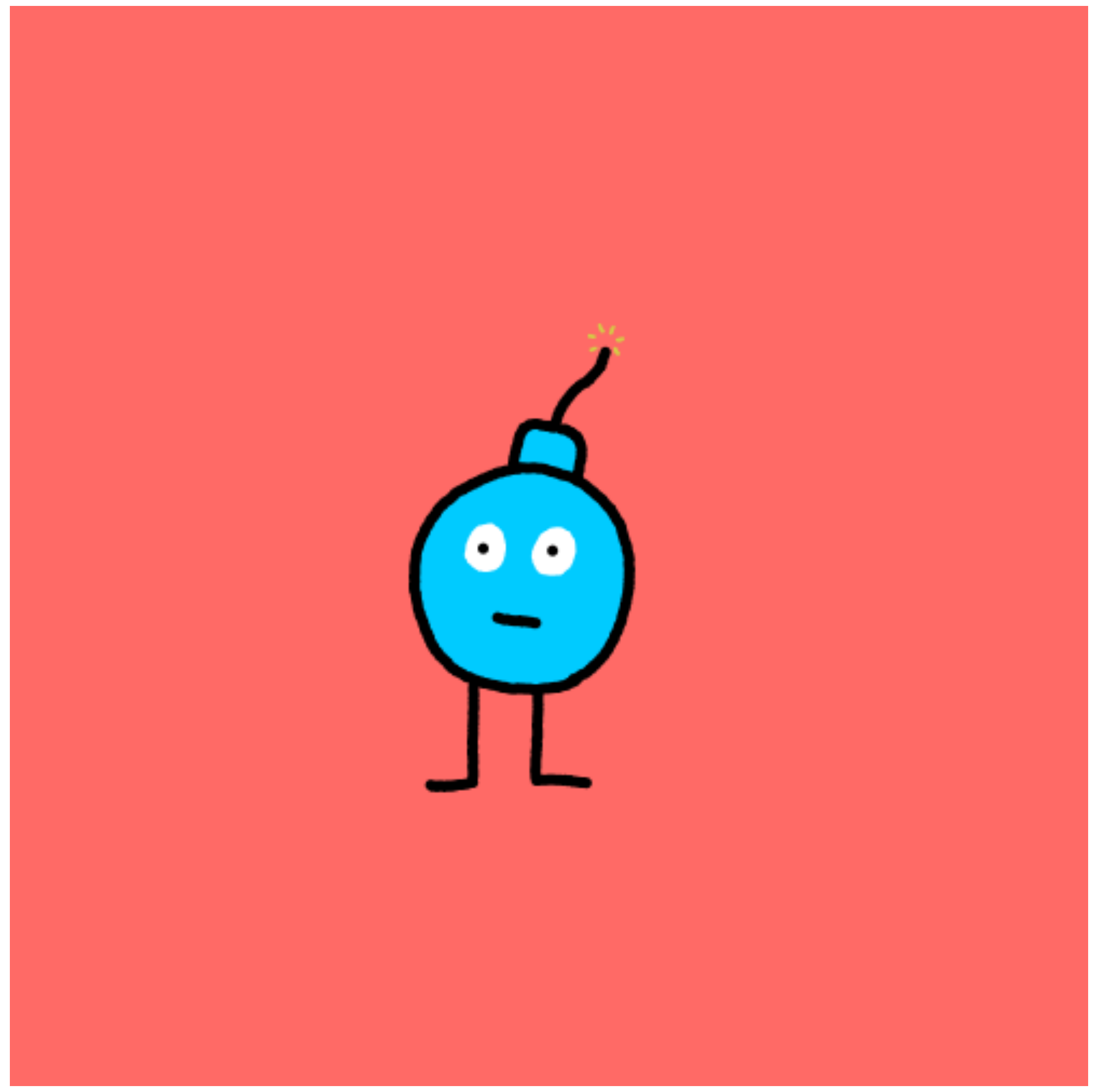


Impact on Cryptography

Shor’s Quantum Algorithm

Polynomial-time algorithm for the following problems:

- Computes **p** given **n** = **pq** → RSA dead
- Computes **d** given **y** = $\mathbf{x}^{\mathbf{d}}$ mod **p** → ECC/DH dead



Practically impossible on a classical machine

#QuantumSpeedup

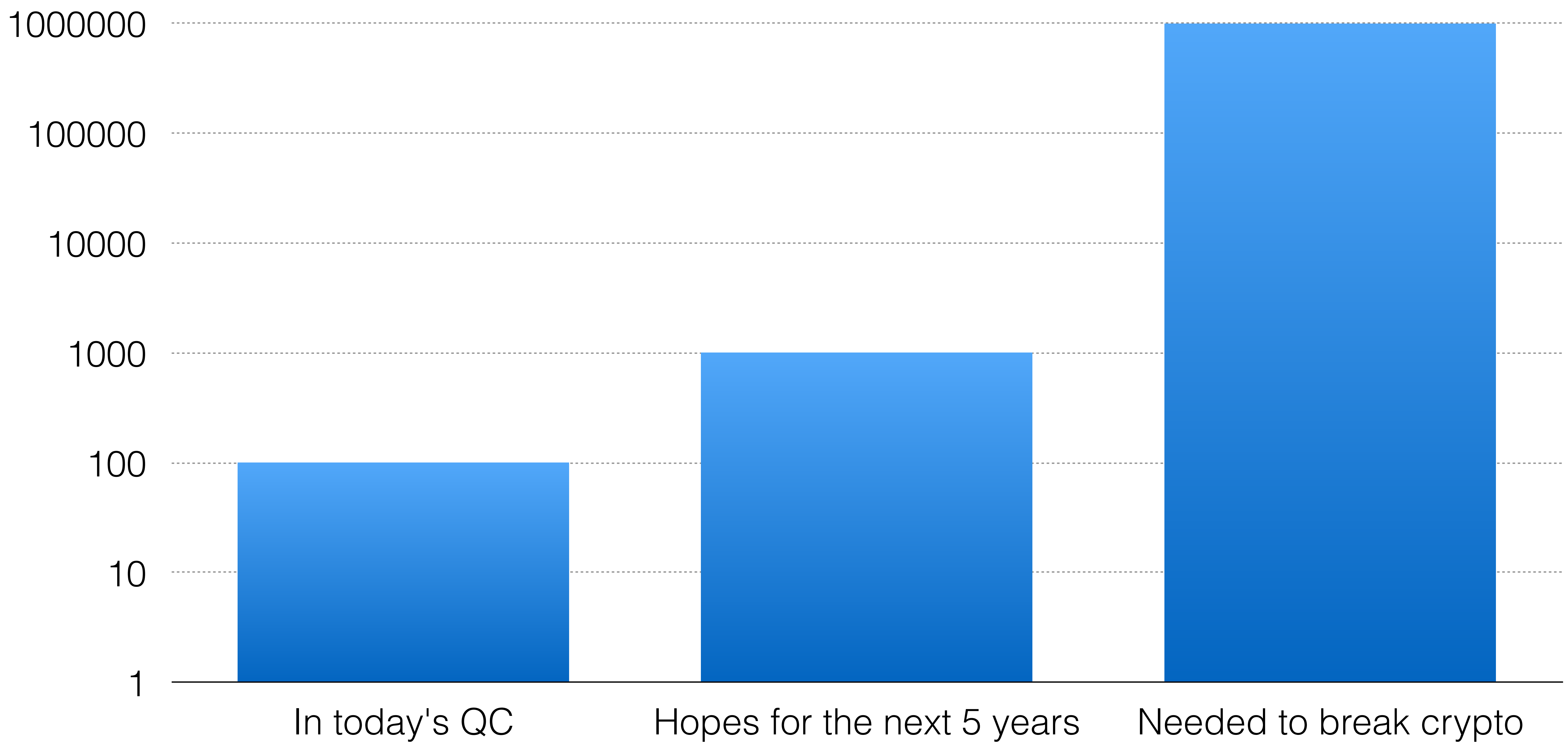
How Bad for Crypto?

Not terrible: Signatures (ECDSA, Ed25519, etc.)
Can be reissued with a post-quantum algorithm
Applications: Bitcoin, application signing

Bad: Key agreement (Diffie-Hellman, ECDH, etc.)
Partially Mitigated by secret internal states and reseeding
Applications: TLS, end-to-end messaging

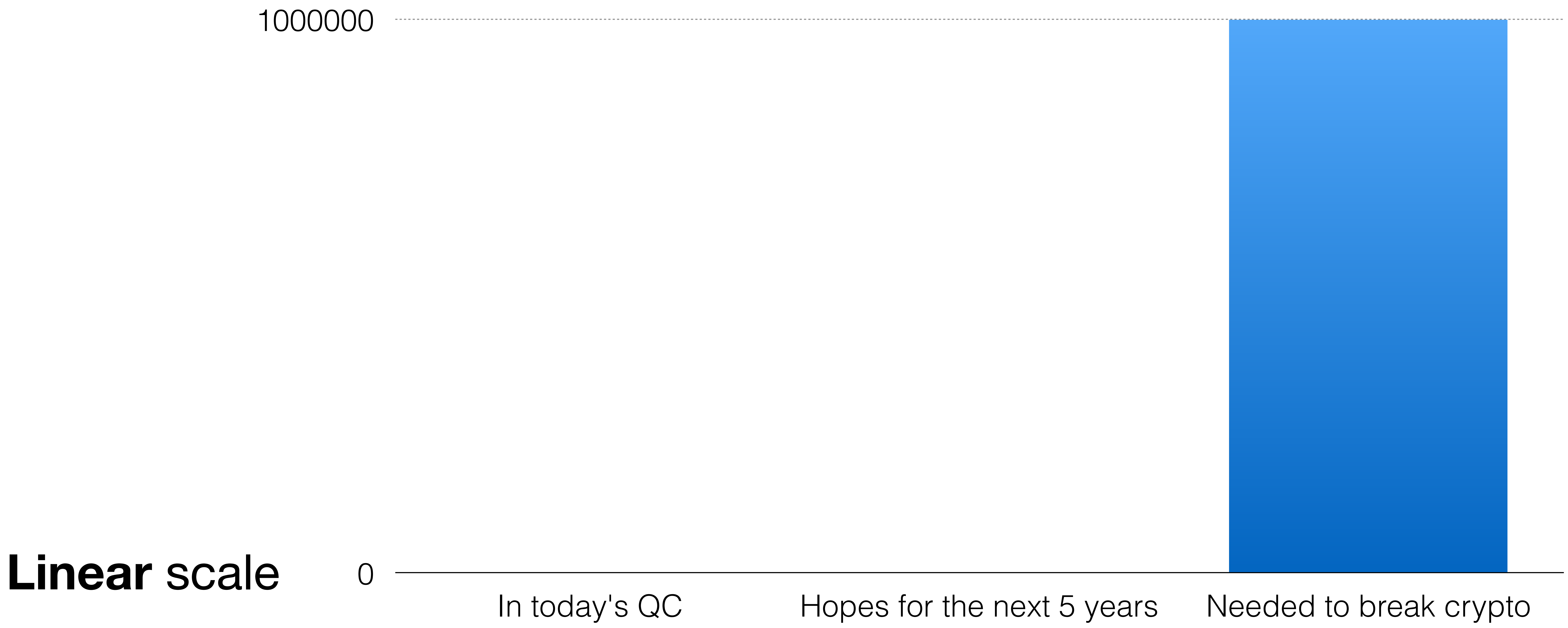
Very annoying: Encryption (RSA encryption, ECIES, etc.)
Encrypted messages compromised forever
Applications: Key encapsulation, secure enclaves

How Many Qubits

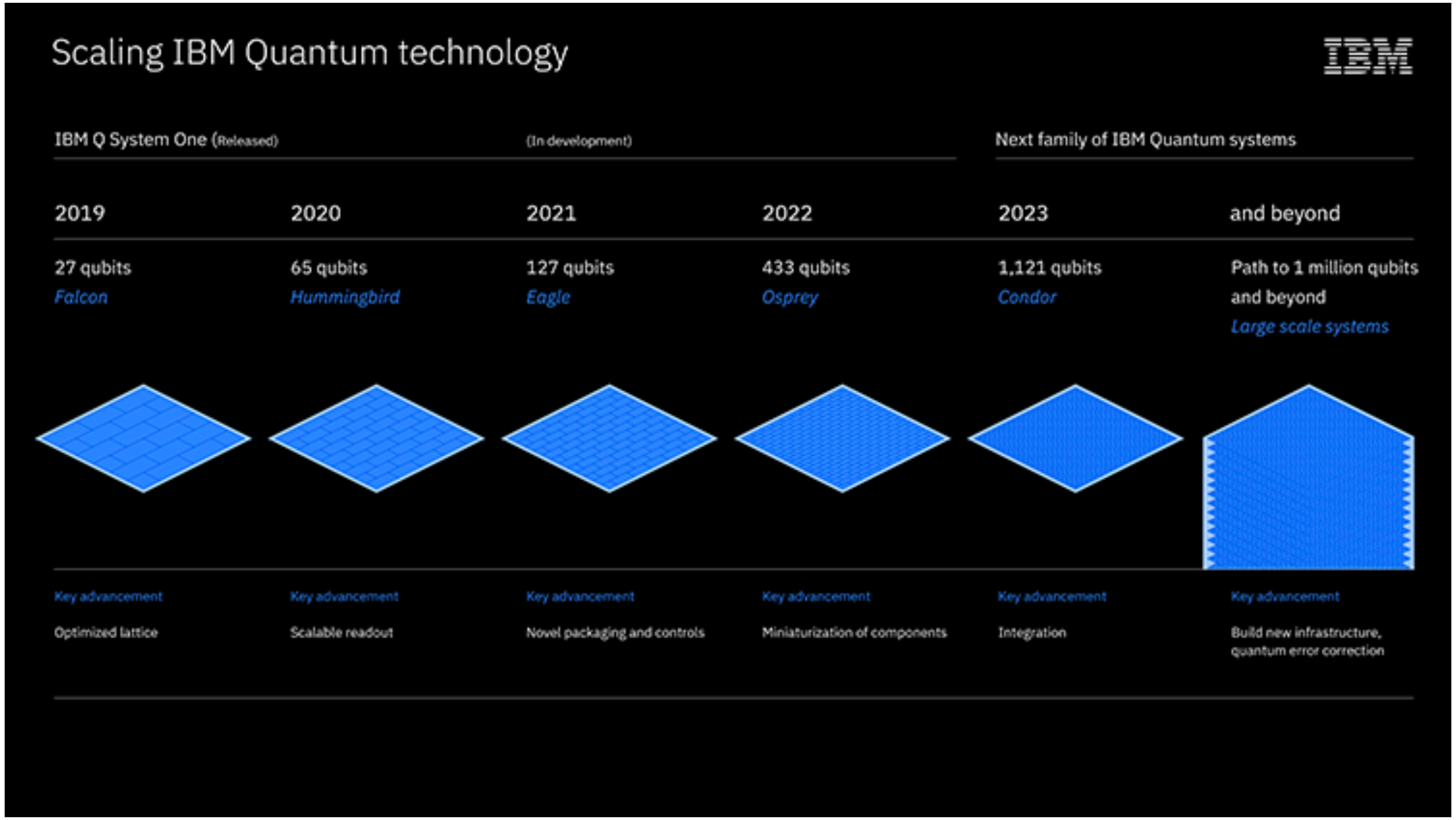


Log scale

How Many Qubits



Quantum Computers Today



PS: “and beyond” might be in a long time, if ever :)

Is D-Wave a Threat to Crypto?

No, because it's not even quantum computing

- Quantum version of simulated annealing, with no evidence of quantum speed-up
- Dedicated hardware for specific optimization problems
- **Can't run Shor**, so can't break crypto. ¯_(ツ)_/¯

Speculative Estimates…

Designing a Million-Qubit Quantum Computer Using Resource Performance Simulator

Muhammad Ahsan, Rodney Van Meter, Jungsang Kim

(Submitted on 2 Dec 2015)

The optimal design of a fault-tolerant quantum computer involves finding an appropriate balance between the burden of large-scale integration of noisy components and the load of improving the reliability of hardware technology. This balance can be evaluated by quantitatively modeling the execution of quantum logic operations on a realistic quantum hardware containing limited computational resources. In this work, we report a complete performance simulation software tool capable of (1) searching the hardware design space by varying resource architecture and technology parameters, (2) synthesizing and scheduling fault-tolerant quantum algorithm within the hardware constraints, (3) quantifying the performance metrics such as the execution time and the failure probability of the algorithm, and (4) analyzing the breakdown of these metrics to highlight the performance bottlenecks and visualizing resource utilization to evaluate the adequacy of the chosen design. Using this tool we investigate a vast design space for implementing key building blocks of Shor's algorithm to factor a 1,024-bit number with a baseline budget of 1.5 million qubits. We show that a trapped-ion quantum computer designed with twice as many qubits and one-tenth of the baseline infidelity of the communication channel can factor a 2,048-bit integer in less than five months.

Speculative Estimates…

"Predicting" quantum computers is a Bayesian game; too little information to make reliable guesses (10 scientists = 12 different predictions)

The Present and Future of Discrete Logarithm Problems on Noisy Quantum Computers

YOSHINORI AONO[1], SITONG LIU[2], TOMOKI TANAKA[3,5], SHUMPEI UNO[4,5], RODNEY VAN METER[2,5] (Senior Member, IEEE), NAOYUKI SHINOHARA[1], RYO NOJIMA[1]

scenario. Their prediction is based on their quantifier of quantum devices that they named generalized logical qubits. They predicted that a superconducting quantum device capable of solving RSA-2048 (using 4,100 qubits) would be available in the early 2050s, rather than before 2039. This is more optimistic than expert opinions [38], [39] published in 2019 and updated in 2020. Mosca and Piani say that 90% of experts predict that there is 50% or greater chance of a quantum device that can break RSA-2048 in 24 hours being released in the next 20 years.

When it Looks too Good to be True..

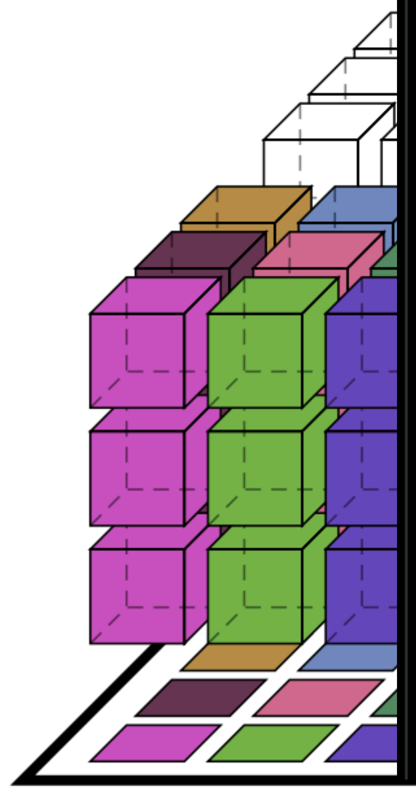
Factoring 2 048 RSA integers in 177 days with 13 436 qubits and a multimode memory

Élie Gouzien* and Nicolas Sangouard†
Université Paris–Saclay, CEA, CNRS, Institut de physique théorique, 91 191 Gif-sur-Yvette, France
(Dated: March 11, 2021)

We analyze the performance of a quantum computer architecture combining a small processor and a storage unit. By focusing on integer factorization, we show a reduction by several orders of magnitude of the number of processing qubits compared to a standard architecture using a planar grid of qubits with nearest-neighbor connectivity. This is achieved by taking benefit of a temporally and spatially multiplexed memory to store the qubit states between processing steps. Concretely, for a characteristic physical gate error rate of 10^{-3}, a processor cycle time of 1 microsecond, factoring a 2 048 bits RSA integer is shown possible in 177 days with a processor made with 13 436 physical qubits and a multimode memory with 2 hours storage time. By inserting additional error-correction steps, storage times of 1 second are shown to be sufficient at the cost of increasing the runtime by about 23 %. Shorter runtimes (and storage times) are achievable by increasing the number of qubits in the processing unit. We suggest realizing such an architecture using a microwave interface between a processor made with superconducting qubits and a multiplexed memory using the principle of photon echo in solids doped with rare-earth ions.

Introduction — Superconducting qubits form the building blocks of one of the most advanced platforms for realizing quantum computers [1]. The standard architecture consists in laying superconducting qubits in a 2D grid and making the computation using only neighboring interactions. Recent estimations showed however that fault-tolerant realizations of various quantum algorithms with this architecture would require millions physical qubits [2–4]. These performance analyses naturally raise the question of an architecture better exploiting the potential of superconducting qubits.

In developing a quantum computer architecture we



Processor


Sam Jaques
@sejaques
Replying to @veorq
Very important caveat: it needs 430 million "memory qubits"

Craig Gidney @CraigGidney · Mar 15
Replying to @quantumVerd @KikeSolanoPhys and 4 others
The paper uses a cost model where quantum memory is comparatively cheap. I'd have included the mem qubit count in the title (at n=2048 there's 13K compute qubits and 430M mem qubits) but don't see anything wrong with considering a world where mem ends up cheaper than cpu.

Quantum Search

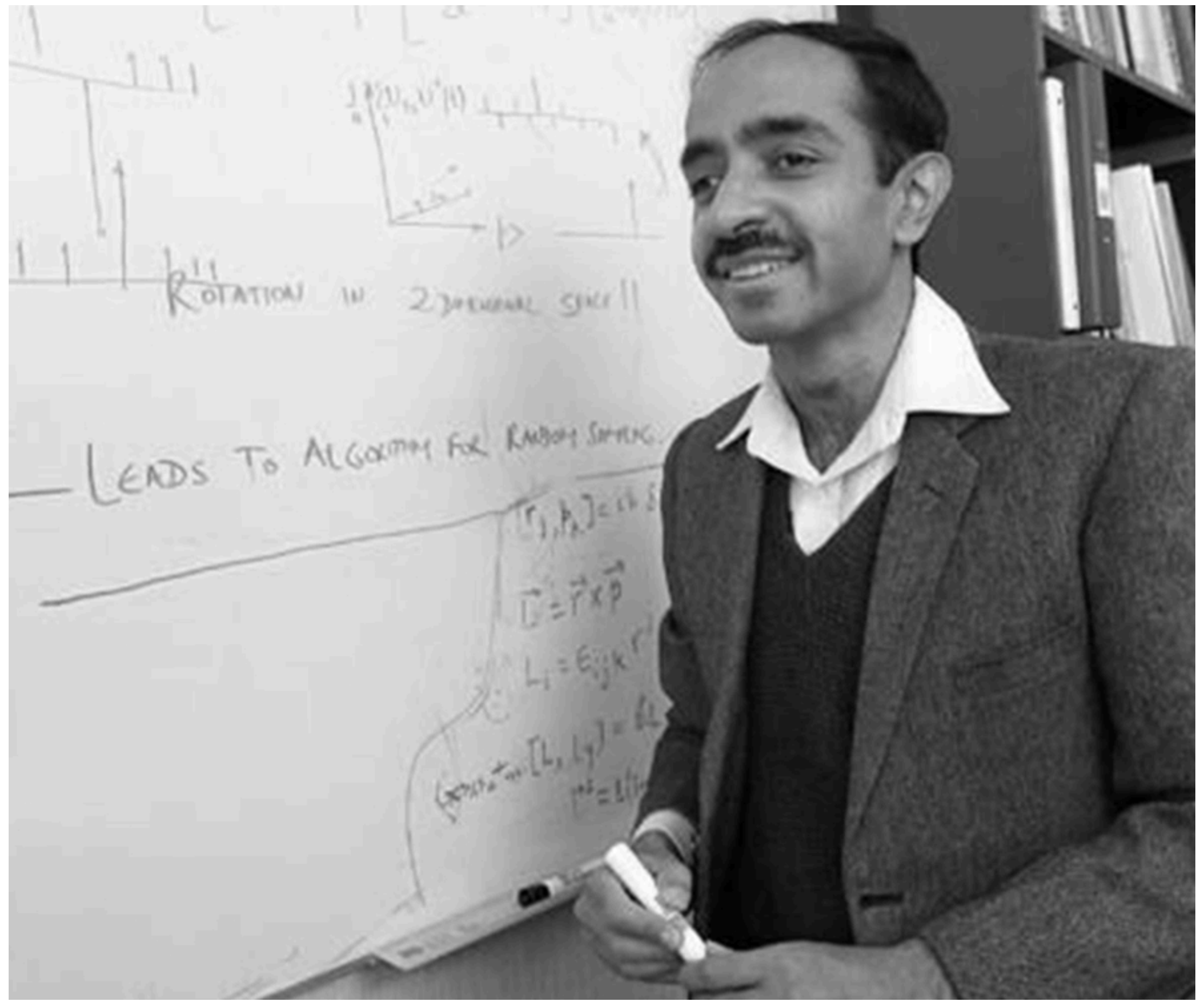
Grover’s algorithm (1996)

Searches in N items in √N queries!

AES-128 broken in $\sqrt{(2^{128})} = 2^{64}$ operations?

Caveats behind this simplistic view:

- Constant factor in **O(√N)** may be huge
- Doesn’t easily parallelise, as classical search does



Quantum-Searching AES Keys

	#gates		depth		#qubits
k	T	Clifford	T	overall	
128	$1.19 \cdot 2^{86}$	$1.55 \cdot 2^{86}$	$1.06 \cdot 2^{80}$	$1.16 \cdot 2^{81}$	2,953
192	$1.81 \cdot 2^{118}$	$1.17 \cdot 2^{119}$	$1.21 \cdot 2^{112}$	$1.33 \cdot 2^{113}$	4,449
256	$1.41 \cdot 2^{151}$	$1.83 \cdot 2^{151}$	$1.44 \cdot 2^{144}$	$1.57 \cdot 2^{145}$	6,681

Table 5. Quantum resource estimates for Grover's algorithm to attack AES-k, where $k \in \{128, 192, 256\}$.

https://arxiv.org/pdf/1512.04965v1.pdf

If gates are the size of a hydrogen atom (12pm) this depth is the **diameter of the solar system** (~10^{13}m), yet less than 5 grams

No doubt more efficient circuits will be designed…

Quantum-Searching AES Keys

From February 2020, better circuits found

Implementing Grover oracles for quantum key search on AES and LowMC

Samuel Jaques[1*†], Michael Naehrig[2], Martin Roetteler[3], and Fernando Virdia[4†‡]

scheme	r	#Clifford	$\#T$	$\#M$	T-depth	full depth	width	G-cost	DW-cost	p_s
AES-128	1	$1.13 \cdot 2^{82}$	$1.32 \cdot 2^{79}$	$1.32 \cdot 2^{77}$	$1.48 \cdot 2^{70}$	$1.08 \cdot 2^{75}$	1665	$1.33 \cdot 2^{82}$	$1.76 \cdot 2^{85}$	$1/e$
AES-128	2	$1.13 \cdot 2^{83}$	$1.32 \cdot 2^{80}$	$1.32 \cdot 2^{78}$	$1.48 \cdot 2^{70}$	$1.08 \cdot 2^{75}$	3329	$1.34 \cdot 2^{83}$	$1.75 \cdot 2^{86}$	1
AES-192	2	$1.27 \cdot 2^{115}$	$1.47 \cdot 2^{112}$	$1.47 \cdot 2^{110}$	$1.47 \cdot 2^{102}$	$1.14 \cdot 2^{107}$	3969	$1.50 \cdot 2^{115}$	$1.11 \cdot 2^{119}$	1
AES-256	2	$1.56 \cdot 2^{147}$	$1.81 \cdot 2^{144}$	$1.81 \cdot 2^{142}$	$1.55 \cdot 2^{134}$	$1.29 \cdot 2^{139}$	4609	$1.84 \cdot 2^{147}$	$1.45 \cdot 2^{151}$	$1/e$
AES-256	3	$1.17 \cdot 2^{148}$	$1.36 \cdot 2^{145}$	$1.36 \cdot 2^{143}$	$1.55 \cdot 2^{134}$	$1.28 \cdot 2^{139}$	6913	$1.38 \cdot 2^{148}$	$1.08 \cdot 2^{152}$	1

Eliminating the Problem: 256-bit Keys

Defeating Quantum Algorithms

A.k.a. “quantum-safe”, “quantum-resilient”

- Must not rely on factoring or discrete log problems
- Must be well-understood with respect to quantum

Why Bother?

Insurance against QC threat:

- “QC has a probability *p* work in year X and the impact would be $N for us”
- “I’d like to eliminate this risk and I’m ready to spend $M for it”

Supposedly the motivation of USG/NSA:

"*we anticipate a need to shift to quantum-resistant cryptography in the near future.”* — NSA in CNSS advisory 02-2015

NSA's Take (Aug 2021)

Q: Is NSA worried about the threat posed by a potential quantum computer because a CRQC exists?
A: NSA does not know when or even if a quantum computer of sufficient size and power to exploit public key cryptography (a CRQC) will exist.

Q: Why does NSA care about quantum computing today? Isn't quantum computing a long way off?
A: The cryptographic systems that NSA produces, certifies, and supports often have very long lifecycles. NSA has to produce requirements today for systems that will be used for many decades in the future, and data protected by these systems will still require cryptographic protection for decades after these solutions are replaced. There is growing research in the area of quantum computing, and global interest in its pursuit have provoked NSA to ensure the enduring protection of NSS by encouraging the development of post-quantum cryptographic standards and planning for an eventual transition.

Q: What are the timeframes in NSS for deployment of new algorithms, use of equipment, and national security information intelligence value?
A: New cryptography can take 20 years or more to be fully deployed to all National Security Systems. NSS equipment is often used for decades after deployment. National security information intelligence value varies depending on classification, sensitivity, and subject, but it can require protection for many decades.

https://media.defense.gov/2021/Aug/04/2002821837/-1/-1/1/Quantum_FAQs_20210804.pdf

"Hey NIST we Need Crypto Standards"

CSRC HOME > GROUPS > CT > POST-QUANTUM CRYPTOGRAPHY PROJECT

POST-QUANTUM CRYPTO PROJECT

NEWS -- August 2, 2016: The National Institute of Standards and Technology (NIST) is requesting comments on a new process to solicit, evaluate, and standardize one or more quantum-resistant public-key cryptographic algorithms. Please see the Post-Quantum Cryptography Standardization menu at left.

Fall 2016	Formal Call for Proposals
Nov 2017	Deadline for submissions
Early 2018	Workshop - Submitter's Presentations
3-5 years	Analysis Phase - NIST will report findings *1-2 workshops during this phase*
2 years later	Draft Standards ready

Finalists

Type	*PKE/KEM*	*Signature*
Lattice[a]	• CRYSTALS-Kyber • NTRU • SABER	• CRYSTALS-Dilithium • FALCON
Code-based	• Classic McEliece	
Multivariate		• Rainbow

“Hey NIST we Need Crypto Standards”

Finalists

Type	PKE/KEM	Signature
Lattice[a]	• CRYSTALS-Kyber • NTRU • SABER	• CRYSTALS-Dilithium • FALCON
Code-based	• Classic McEliece	
Multivariate		• Rainbow

'Moody, Dustin (Fed)' via pqc-forum Tue, Apr 19, 10:09 PM (11 days ago)
to Dustin, pqc-forum

Everybody,

We appreciate your patience. The announcement of the algorithms we will standardize is still coming very soon. This is a major milestone of our project, and the delay is not due to technical considerations but is due to some legal and procedural steps that are taking more time than we anticipated. Again, thank you for your patience.

The PQC team

The Five Families

- Based on **coding theory** (McEliece, Niederreiter):
  - Solid foundations from the late 1970s, large keys, encryption only
- Based on **multivariate polynomials** evaluation
  - Based on multivariate equations' hardness, mostly for signatures
- Based on **hash functions** and **tree-based** constructions
  - Ideas from the 70s, as secure as the hash, large keys, signature only
- Based on **elliptic curve isogenies**
  - More recent problem, relatively slow, Diffie-Hellman-like key agreement
- Based on **lattice problems…**

Lattice-Based Crypto: Intuition

Based on problems such as **learning with errors** (LWE):

- **S** a secret vector of numbers modulo q
- Receive pairs (**A**, **B**)
  - **A** = (A_0, …, A_{n-1}) is a vector of uniformly random numbers
  - **B** = <**S**, **A**> + **E** , a vector of B_i = $S_i * A_i + E_i$
  - **E** = (E_0, …, E_{n-1}) is an **unknown** vector or *normal*-random numbers

Attacker goal: find **S** given many pairs (**A**, **B**)

Without **E**: trivial (linear systems of equations)

With **E**: NP-**hard**

Lattice-Based Crypto: Intuition

The best balance between performance and security assurance

Heated discussions about their relative merits, and speculative theories..

NIST Round 3

Code-based:

- Classic McEliece (KEM, finalist)

Lattice-based:

- Dilithium (signature, finalist)
- Falcon (signature, finalist)
- Kyber (KEM, finalist)
- NTRU (KEM, finalist)
- SABER (KEM, finalist)

MQ-based:

- Rainbow (signature, finalist)

Kyber's inefficiency: some data points 657 views

D. J. Bernstein Oct 20, 2021,

to pqc-...@list.nist.gov

Some recent comments seem to assume that Kyber is the most efficient lattice KEM in NISTPQC. However, if I want an ARM Cortex-M4 to decrypt messages, specifically with Core-SVP >= 2^128, then my costs (sorted by cycles+1000*bytes) are as follows, according to (1) pqm4 benchmarks from https://github.com/mupq/pqm4/blob/master/benchmarks.md, (2) tables of cip

cpei...@alum.mit.edu Oct 30, 2021, 11:41:37 PM

to pqc-forum

Dan, in your long reply, I didn't see a straight answer to my question about the algorithmic claim from the end of your talk of 10+ weeks ago, namely:

"Heuristics imply [Hermite factor] <= n^{1/2+o(1)} in time exp(n^{1/2+o(1)})" for cyclotomic Ideal-SVP.

As I see it, the central question is this: is this claim still in effect, and if so, when can the community expect to see it substantiated?

For such a remarkable claim, one would normally expect to see a research paper backing it up, and before so much time has passed.

[CFRG] NSA vs. hybrid

D. J. Bernstein via ietf.org Fri, Nov

to cfrg

This looks to me like something that should be discussed in CFRG rather than LAMPS:

https://datatracker.ietf.org/meeting/112/materials/slides-112-lamps-hybrid-no
https://mailarchive.ietf.org/arch/msg/spasm/McksDhejGgJJ6xG617FEWLbB

This is one part of a big push by NSA across multiple non-CFRG venues to convince everyone to

* deploy small lattice systems---which _hopefully_ protects against quantum computers---and

* _turn off ECC_---this is the scary part, since there's a serious risk that the small lattice systems are easier to break than ECC.

See analyses at https://ntruprime.cr.yp.to/warnings.html

PQC Performance

Algorithm	Public key (bytes)	Ciphertext (bytes)	Key gen. (ms)	Encaps. (ms)	Decaps. (ms)	
ECDH NIST P-256	64	64	0.072	0.072	0.072	Elliptic curves (not post-quantum)
SIKE p434	330	346	13.763	22.120	23.734	Isogeny-based
Kyber512-90s	800	736	0.007	0.009	0.006	Lattice-based
FrodoKEM-640-AES	9,616	9,720	1.929	1.048	1.064	Lattice-based

Table 1: Key exchange algorithm communication size and runtime

Algorithm	Public key (bytes)	Signature (bytes)	Sign (ms)	Verify (ms)	
ECDSA NIST P-256	64	64	0.031	0.096	
Dilithium2	1,184	2,044	0.050	0.036	Lattice-based
qTESLA-P-I	14,880	2,592	1.055	0.312	Lattice-based
Picnic-L1-FS	33	34,036	3.429	2.584	Zero-knowledge proof-based

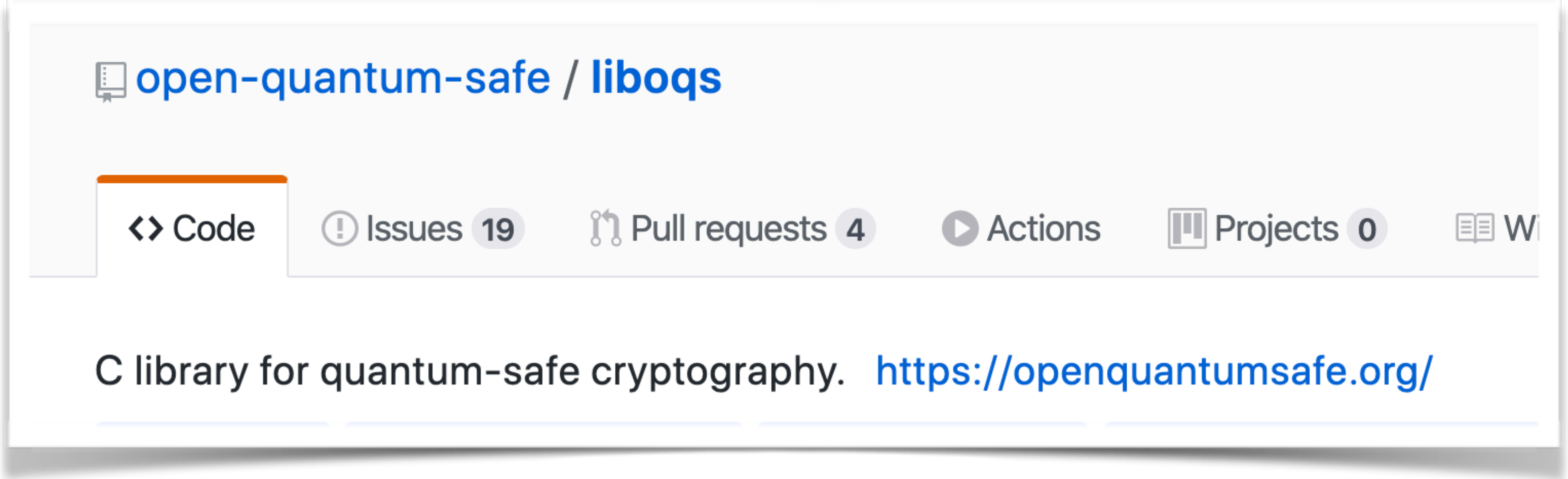
Table 2: Signature scheme communication size and runtime

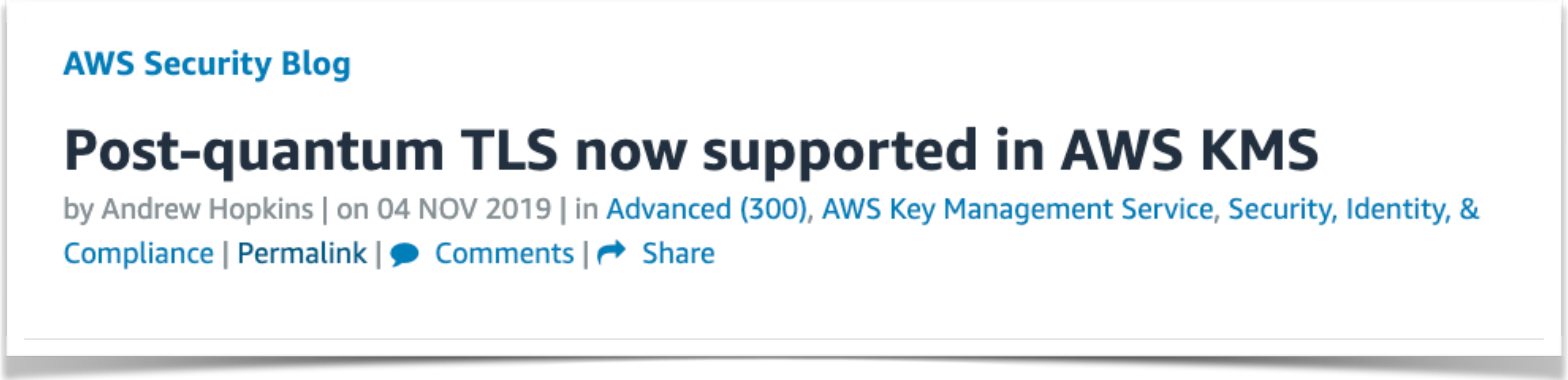
From "Benchmarking Post-Quantum Cryptography in TLS” https://eprint.iacr.org/2019/1447

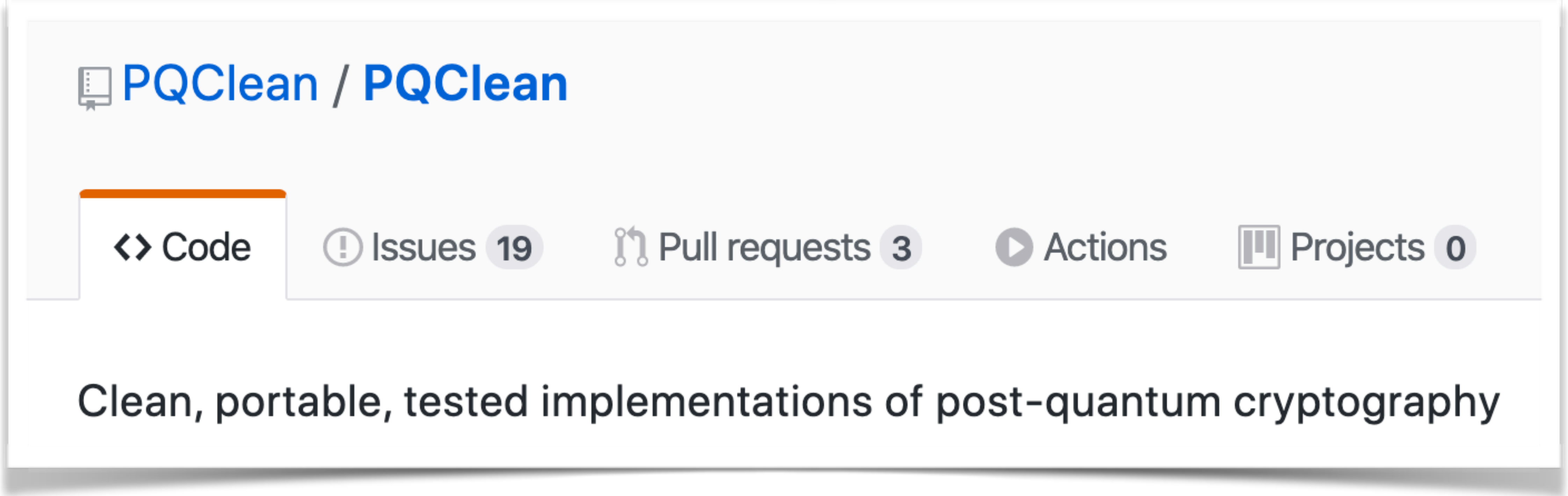
Using PQC Today

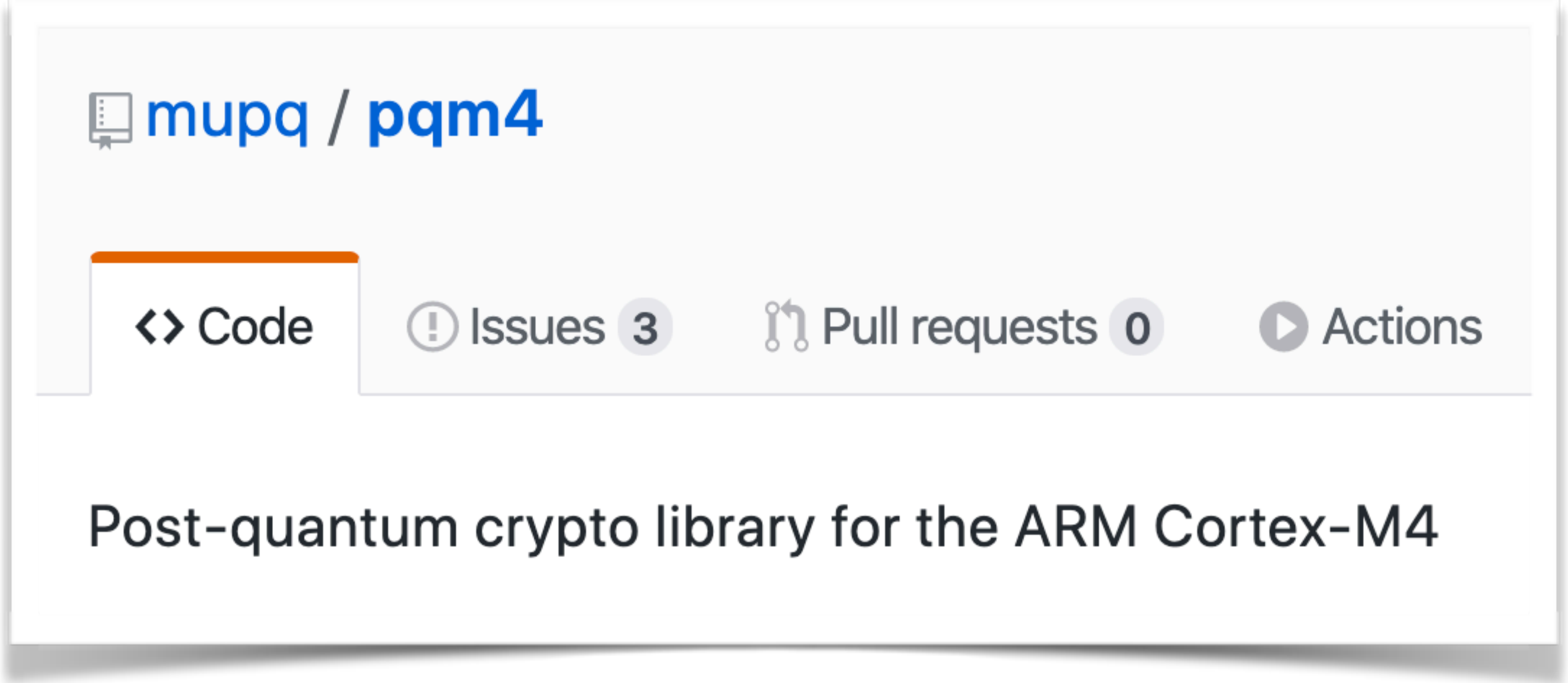
Libraries, implementations, specifications (for TLS, IPsec), standards

See https://github.com/veorq/awesome-post-quantum









TAURUS

Thank you

jp@taurusgroup.ch