Quantum Computing and Post-quantum Cryptography
PART 1

André Schrottenloher

Inria Rennes
Team CAPSULE

The quantum menace

The race to save the Internet from quantum hackers

Researchers have created a new and potentially dangerous encryption-breaking quantum algorithm

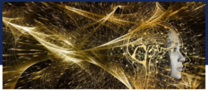
Some cryptography researchers see the claim as misleading, others see it as a potential warning sign

By Jimmy Pezzone January 14, 2023 at 2:38 PM | 23 comments

IBM: Quantum computing poses an 'existential threat' to data encryption

Tim Keary
@tim_keary

January 17, 202



Quantum computers will crack your encryption—maybe they already have

Research teams worldwide are racing to create a computer so powerful it will be able to read encrypted messages.

Will quantum computers break RSA encryption in 2023?

Everybody knows that we should prepare ourselves for a "quantum future", but it was expected to come about in 10-20 years' time. Is a breakthrough possible this year?

Quantum computers can break major encryption method, researchers claim

It has long been known that one day quantum computers will probably be able to crack the RSA encryption method we use to keep data safe, but a team of researchers is now claiming it is already possible, while others say the results require more scrutiny

Have Chinese scientists really cracked RSA encryption with a quantum computer?

Outline

Quantum Computing Basics

Today: the Talk

See the comic by Scott Aaronson & Zach Weinersmith at:
https://www.smbc-comics.com/comic/the-talk-3

Brief summary of quantum physics

I think I can safely say that nobody understands quantum mechanics.

– Richard Feynman (1918-1988)

- interpreting quantum physics is difficult
- good for us: we're not here to interpret, just to calculate

Brief history of quantum computing

- **Quantum computing** initiated in the 80s with the prospect of **simulating quantum mechanical systems**

$\Longrightarrow$ e.g., to understand protein folding

- Could it also be used to speed up **classical computations?**

$\Longrightarrow$ first significant quantum speedups appeared in the 90s

---

Deutsch, "Quantum theory, the Church-Turing principle and the universal quantum computer", Proc. R. Soc. Lond. 1985

Qubits and superposition

A **bit** is a classical system which can be in the state 0 **or** 1.

$$b = 0 \text{ or } 1$$

A **qubit** is a quantum system with two **basis states** $|0\rangle$ **and** $|1\rangle$.

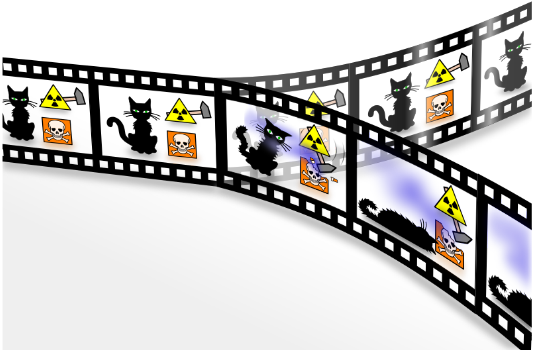
$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

α and β are complex numbers such that $|\alpha|^2 + |\beta|^2 = 1$

Measurement

- The state is a **superposition**
- **Measuring** the qubit **destroys** the state and **collapses** the superposition to $|0\rangle$ or $|1\rangle$
- $|0\rangle$ is measured with probability $|\alpha|^2$
- $|1\rangle$ is measured with probability $|\beta|^2$

Qubits and superposition (ctd.)



$$|\text{cat}\rangle = \frac{1}{\sqrt{2}}\,|\text{ cat is alive}\rangle + \frac{1}{\sqrt{2}}\,|\text{ cat is dead}\rangle$$

- any two-state quantum system can be used as a qubit: even a cat

Qubits and entanglement

- **Two bits** can be in the state 00 **or** 01 **or** 10 **or** 11.
- **Two qubits** form a quantum system with **4 basis states** $|00\rangle, |01\rangle, |10\rangle, |11\rangle$

(4-dimensional vector space)

Consider the following state:

$$|\psi\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Measure the first qubit: the second **always collapses** to $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

$\Longrightarrow$ the two qubits are **disentangled**

Qubits and entanglement (ctd.)

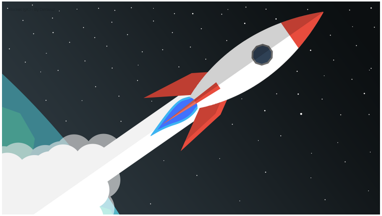
Consider the following state:

$$|\psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

Measure the first qubit:

- if the state **collapses** to $|00\rangle$: we measure 0 and the other becomes 0 with certainty
- if the state collapses to $|11\rangle$: we measure 1 and the other is 1 with certainty

Qubits and entanglement (ctd.)



- It still works if you send the second qubit to space: its state will collapse on 0 or 1 **depending on the measurement result**

- Experiments in the 1980s confirmed the theory

- Unfortunately for sci-fi, this doesn't allow faster-than-light communication

Picture: École polytechnique / Jérémy Barande

Qubits and entanglement (ctd.)

n **qubits** form a 2^n-dimensional quantum system with 2^n basis states:

$$|\psi\rangle = \alpha_{00..0}|00..0\rangle + \alpha_{01..0}|01..0\rangle + \ldots + \alpha_{11..1}|11..1\rangle \in \mathbb{C}^{2^n}$$

It is (and remains) normalized: $\sum_i |\alpha_i|^2 = 1$.

An n-qubit quantum system is described by 2^n **complex amplitudes**. If the system evolves, we must recompute the 2^n amplitudes.

- this gets rapidly out of hand for classical computers
- this is why quantum computers were proposed in the first place!

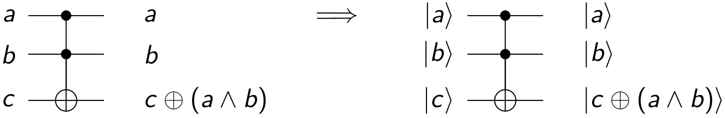
Computations

- We start from a set of qubits initialized to $|00...0\rangle$
- We describe quantum algorithms as a sequence of basic, elementary **quantum gates**
- The quantum gates modify the current state of the algorithm
- Eventually we will **measure** the state

Starting from classical circuits

- Any **classical (reversible)** circuit can be applied to our qubits
- It will just apply **in superposition** to all possible states

Example: the **Toffoli gate** (universal for reversible logic)



$$|001\rangle \to |001\rangle\,, \qquad |111\rangle \to |110\rangle$$

$$\frac{1}{\sqrt{2}}|001\rangle + \frac{1}{\sqrt{2}}|111\rangle \to \frac{1}{\sqrt{2}}|001\rangle + \frac{1}{\sqrt{2}}|110\rangle$$

A quantum computation is a **linear operator**.

Adding more quantum operations

The one-qubit **Hadamard gate**:

$$H|0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle, \qquad H|1\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

The n-qubit **Hadamard transform**:

$$\forall x, H|x\rangle = \frac{1}{\sqrt{2^{n}}}\sum_{y}(-1)^{x\cdot y}|y\rangle$$

A quantum computation is a **unitary operator** (to preserve the normalization).

"Quantum parallelism"

Let $f : \{0,1\}^n \to \{0,1\}^m$ be your favorite function (e.g., SHA-3). There exists a reversible circuit doing:

$$x, 0 \mapsto x, f(x)$$

i.e. a quantum algorithm:

$$|x\rangle |0\rangle \mapsto |x\rangle |f(x)\rangle$$

Start from a uniform superposition over x:

$$\left(\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \right) |0\rangle$$

apply f:

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle |f(x)\rangle$$

"Quantum parallelism" (ctd.)

$$\frac{1}{\sqrt{2^{n}}} \sum_{x \in \{0,1\}^{n}} |x\rangle \, |f(x)\rangle$$

Let's say we want a preimage of f, i.e., we fix y, and want x such that $f(x) = y$. It's here!

Are we simply computing "all the possibilities" in parallel?

NO

In superposition $\neq$ in parallel

If we measure the state, we obtain a random $x, f(x)$: this is useless

Summary

The 3 principles of quantum computing:

1. superposition
2. entanglement
3. **interference** (next slides)

And the 4th one:

- Quantum computation is **not** "doing everything is parallel"

Examples of Quantum Algorithms

Deutsch's algorithm

Consider a one-bit boolean function: $f : \{0,1\} \to \{0,1\}$ implemented by a **phase oracle**: $|x\rangle \mapsto (-1)^{f(x)}|x\rangle$.

Problem: determine if f is **constant** $f(0) = f(1)$ or **balanced** $f(0) \neq f(1)$.

- Classically we need 2 queries
- Quantumly we need **1**

$$|0\rangle \;—\boxed{\mathrm{H}}—\boxed{f}—\boxed{\mathrm{H}}—$$

$$|0\rangle \overset{H}{\mapsto} \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \overset{f}{\mapsto} \frac{1}{\sqrt{2}}(-1)^{f(0)}|0\rangle + \frac{1}{\sqrt{2}}(-1)^{f(1)}|1\rangle \quad .$$

Deutsch's algorithm (ctd.)

- If $f(0) = f(1)$:

$$H\left[\pm\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right] = \pm\frac{1}{2}\left[\underbrace{(|0\rangle + |1\rangle) + (|0\rangle - |1\rangle)}\right] = \pm|0\rangle$$

$\Longrightarrow$ **constructive interference** on 0 and **destructive** on 1

$\Longrightarrow$ measure 0

- If $f(0) \neq f(1)$:

$$H\left[\pm\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\right] = \pm\frac{1}{2}\left[\underbrace{(|0\rangle - |1\rangle) - (|0\rangle - |1\rangle)}\right] = \pm|1\rangle$$

$\Longrightarrow$ **constructive interference** on 1 and **destructive** on 0

$\Longrightarrow$ measure 1

Simon's algorithm

Consider an n-bit (2-to-1) boolean function: $f\ :\{0,1\}^{n} \to \{0,1\}^{n}$

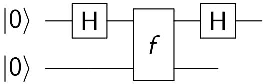
- with a circuit implementation $x, 0 \mapsto x, f(x)$
- with a **hidden boolean period**: $f(x \oplus s) = f(x)$

Problem: find s.

To answer this question, we need $\simeq 2^{n/2}$ queries classically. Simon's algorithm does it in $\simeq n$ queries!

Simon, "On the power of quantum computation", FOCS 1994

Simon's algorithm (ctd.)



$$|0\rangle\,|0\rangle \mapsto \frac{1}{\sqrt{2^{n}}} \sum_{x\in\{0,1\}^{n}} |x\rangle\,|0\rangle \mapsto \frac{1}{\sqrt{2^{n}}} \sum_{x\in\{0,1\}^{n}} |x\rangle\,|f(x)\rangle$$

At this point, measure $f(x)$. We get a random value a. Let $x_0, x_0 \oplus s$ be such that: $f(x_0) = f(x_0 \oplus s) = a$.

$$\frac{1}{\sqrt{2}}\left(|x_0\rangle + |x_0 \oplus s\rangle\right)$$

Now do another H. The states x_0 and $x_0 \oplus s$ will **interfere**:

$$\frac{1}{\sqrt{2^{n+1}}}\left(\sum_{y}(-1)^{y\cdot x_0}\,|y\rangle + \sum_{y}(-1)^{y\cdot(x_0\oplus s)}\,|y\rangle\right)$$

$$= \frac{1}{\sqrt{2^{n+1}}}\left(\sum_{y}(-1)^{x_0\cdot y}\left(1+(-1)^{s\cdot y}\right)|y\rangle\right)$$

Simon's algorithm (ctd.)

In this final state:

- If $s \cdot y = 0$, the amplitude of y is $\frac{1}{\sqrt{2^{n-1}}}$
- If $s \cdot y = 1$, **the amplitude of y is** 0

$\Longrightarrow$ we can only measure y such that $s \cdot y = 0$

$\Longrightarrow$ we obtain (random) linear equations on s

$\Longrightarrow$ we find s after repeating this $\simeq n$ times

Shor's algorithm has a similar structure.
$\oplus$ is replaced by a modular $+$, H by a more general Quantum Fourier Transform.

Most common speedup: Grover's algorithm

Given a large search space:

- you can pick a guess
- you can test if your guess is "good" with a quantum circuit
- the probability to be good is p

$\Longrightarrow$ Grover search runs in time $\simeq \frac{1}{\sqrt{p}}$

An important message

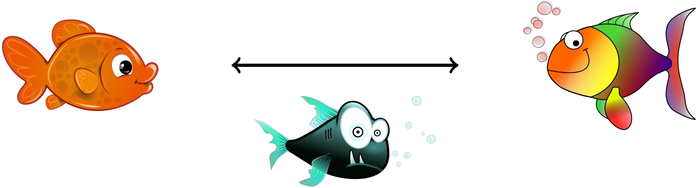
If the test is a **black box**, the quadratic speedup is **optimal**.

$\Longrightarrow$ $\sqrt{\cdot}$ speedup of many NP-complete problems, crypto problems, etc.

Quantum Algorithms vs. Cryptography

Cryptography in a nutshell

Enable (cheap) secure communications over insecure channels.



Public-key

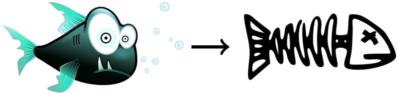
- **No shared secret**
- Key-exchange, signatures. . .
- RSA, elliptic curve cryptography . . .

Secret-key

- **Shared secret**
- Block ciphers, stream ciphers, hash functions. . .
- AES, SHA-3 . . .

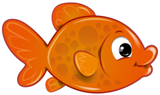
Computational hardness

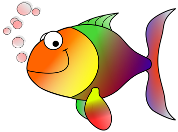
- Cryptography is based on **conjectured** hard computational problems
- To decrypt the communication,  has to factor large numbers, or find a secret AES key, etc.
- We estimate the **time** it would take to reach these goals, and ensure that it's infeasible



Example: Diffie-Hellman key-exchange

- **Alice** and **Bob** want to create a secret key k
- They agree on a large prime p and a multiplicative generator g of $\mathbb{Z}_p^*$





1.	**Alice** chooses $a \in \{1, \ldots, p-1\}$		**Bob** chooses $b \in \{1, \ldots, p-1\}$
2.	**Alice** sends $g^a \mod p$	$\rightarrow$	
3.		$\leftarrow$	**Bob** sends $g^b \mod p$
4.	**Alice** computes $(g^b)^a \mod p$		**Bob** computes $(g^a)^b \mod p$

They define the secret key: $k := (g^a)^b = g^{ab} = (g^b)^a \mod p$.

Diffie-Hellman key-exchange (ctd.)

- Meanwhile,  observes g^{a}, g^{b}, and would like to find g^{ab}

$\Longrightarrow$ This is the **Diffie-Hellman problem**

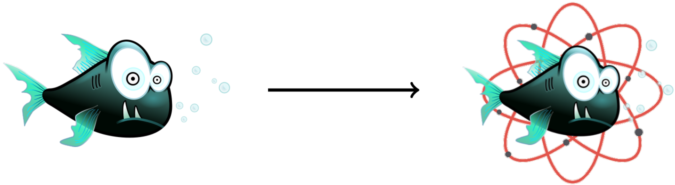
-  could also find a directly from g^{a}

$\Longrightarrow$ this is the **discrete logarithm** problem (DLP)

If the group is chosen well, this problem is **hard** ...

... but not with a quantum computer.

Diffie-Hellman vs. Shor's algorithm (Shor wins)



- The **security** of DH relies on the **hardness** of the DLP
- Similarly the security of RSA relies on the hardness of factoring
- Shor's algorithm solves **factoring and DLP** in polynomial time, by reducing both to **Abelian hidden period**

$\Longrightarrow$ breaks all public-key crypto used today

Shor, "Algorithms for quantum computation: discrete logarithms and factoring", FOCS 1994

Post-quantum cryptography

- Solution: do not use DLP and factoring-based crypto anymore!

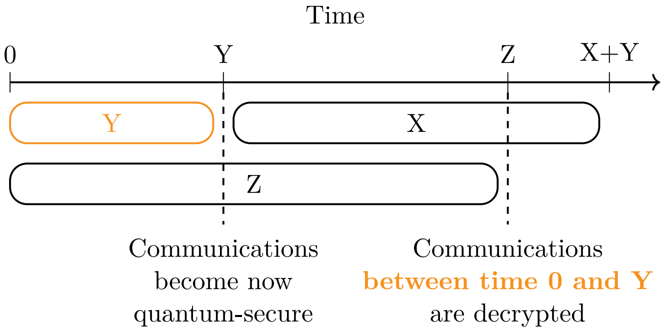
Post-quantum crypto = crypto that remains secure in the presence of a quantum adversary.

"But the quantum computer does not exist yet!"

$\Longrightarrow$ The communication should remain secret for a time X (50 years?)
$\Longrightarrow$ Changing to post-quantum crypto will take time Y (10 years?)
$\Longrightarrow$ Building a QC will take time Z (30 years?)

"Mosca's theorem"

If $Y + X > Z$, you have a problem



"And what if it never works?"

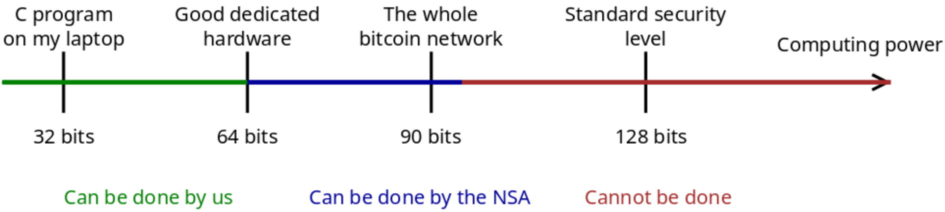
Reversed pascalian bet:

- there is a probability p that the "QC event" happens in our lifetimes
- if this happens, all the technological infrastructure is at risk

Even if $p \to 0$, the risk remains large.

Reasoning about quantum adversaries

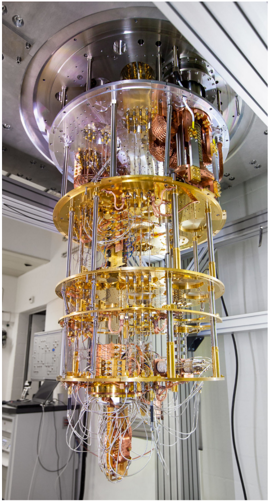
- Crypto is already about **attackers that do not exist**
- Many attacks are **theoretical algorithms** which show a weakness, but which will never run in practice



- Upgrading to PQC means just updating the notion of "algorithm", and the landscape of attacks

Conclusion

Where are we today?

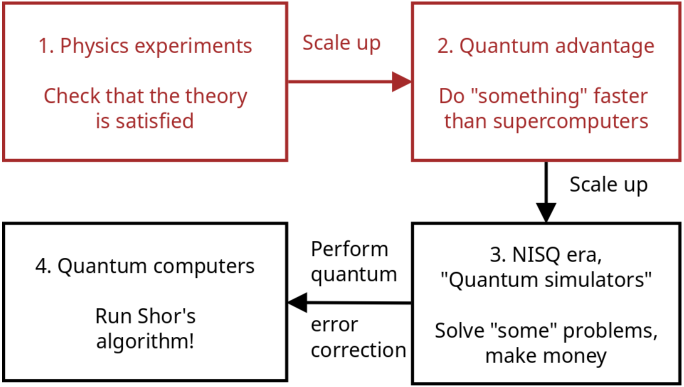


Many competitors, some giants (IBM, Google, Rigetti. . .), lots of start-ups.

- Alice & Bob (cat qubits)
- Pasqal (neutral atoms)
- Quandela (photonics)

Picture : IBM Research

Where are we today? (ctd.)



- Quantum computers are better than classical computers...
- ...at being quantum computers
- ...not on "useful" problems

Where are we today? (ctd.)

Current numbers: $\simeq 10^3 - 10^4$ gates on $\simeq 10^2 - 10^3$ qubits.

$$\vdots$$

What we need: $\simeq 10^9 - 10^{10}$ gates on $\simeq 10^3 - 10^4$ qubits

- But current qubits are "physical": they have lots of errors
- To run large-scale computations, we will need to **correct the errors**

$\Longrightarrow$ either technological breakthroughs, or more scaling

⟨Quantum
the open journal for quantum science

CALL FOR EDITORS! PAPERS PERSPECTIVES

How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney[1] and Martin Ekerå[2,3]

[1]Google Inc., Santa Barbara, California 93117, USA
[2]KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden
[3]Swedish NCSA, Swedish Armed Forces, SE-107 85 Stockholm, Sweden

Published: 2021-04-15, volume 5, page 433

Conclusion

- Quantum computers are **extremely good** at solving some **specific problems** (e.g., Shor)
- ...and very good at solving other (less specific) problems (e.g., Grover)
- ...and totally useless at solving many other problems!

- Quantum computers today are still **experimental**
- ...but making steady progress towards the first use cases
- Still a long way from breaking crypto

Cryptographers have been unlucky with Shor's algorithm, but we're going to make our cryptography **post-quantum**. See part 2!