Quantum Computers vs. Computers Security

@veorq — http://aumasson.jp
Schrodinger equation
Entanglement
Bell states
EPR pairs
Wave functions
Uncertainty principle
Tensor products
Unitary matrices
Hilbert spaces

Nobody understands this stuff, and you don’t need it to understand quantum computing
1. QC 101
2. In practice
3. Breaking crypto
4. Post-quantum crypto
5. Quantum key distribution
6. Quantum copy protection
7. Quantum machine learning
8. Conclusions
1. QC 101
Quantum mechanics

Nature’s operating system

Applications
- Gravity
- Electromagnetism
- Nuclear forces

OS
- Quantum mechanics

Hardware
- Mathematics
Quantum mechanics

Particles in the universe behaves randomly

Their probabilities can be negative

"Negative energies and probabilities should not be considered as nonsense. They are well-defined concepts mathematically, like a negative of money."

—Paul Dirac, 1942
Quantum bit (qubit)

\[ \alpha |0\rangle + \beta |1\rangle \]

When observed

0 with probability \( \alpha^2 \)
1 with probability \( \beta^2 \)

Once observed, stays either 0 or 1 forever
Quantum byte

\[ \alpha_{0x00} |0x00\rangle + \ldots + \alpha_{0xfe} |0xfe\rangle + \alpha_{0xff} |0xff\rangle \]

Again, the sum of probabilities \( \alpha^2 \) equals 1

The \( \alpha \)'s are called amplitudes

Generalizes to 32- or 64-bit quantum words
Quantum computer

Set of quantum registers (bits/bytes/words)

Quantum assembly instructions:
Transform the probabilities of the register
Probabilities should still sum to 1
Linear math transforms (matrix products)

A program ends with a measurement
Quantum computer simulators

Quantum Computing Playground

Quantum Computing Playground is a browser-based quantum computer and quantum circuit simulator. It features a GPU-accelerated quantum circuit simulator, a simple IDE interface, and its own scripting language, QScript. It provides 3D quantum state visualization features. Quantum Computing Playground can efficiently simulate quantum registers up to 50 qubits, Shor's algorithm, and has a variety of quantum physics calculation scripting language itself.

Start with Basic Example »

Play with Shor's Algorithm »

www.quantiki.org/wiki/List_of_QC_simulators

List of QC simulators

Personal tools
- Log in / create account

Content
- Current events
- News
- Jobs
- Groups
- Forums
- Videos
- Bibliography
- About Quantiki

Wiki Navigation
- Main Page
The killer app

Simulating Physics with Computers

Richard P. Feynman

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

Received May 7, 1981

Impossible with a classical computer

Possible with a quantum computer!
QC vs. hard problems

You heard about **NP-complete** problems? SAT, scheduling, Candy Crush, etc.
Solution hard to find, but easy to verify

**QC does not** solve NP-complete problems!

BQP (quantum)

<table>
<thead>
<tr>
<th></th>
<th>NP</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>hard</strong></td>
<td></td>
<td><strong>easy</strong></td>
</tr>
</tbody>
</table>
Quantum speedup

Make the impossible possible

Example: **Factoring integers**

Hard classically (exponential-ish)

Easy with a quantum computer!

Obvious application: **break RSA!**
Quantum parallelism

“Qubits encode all values at the same time!”

Caveat: you can only observe one result

Different observations in different worlds
2. In practice
Factoring experiments

143 is largest number yet to be factored by a quantum algorithm

Quantum factorization of 56153 with only 4 qubits

Only for numbers with special patterns

Not really the real thing (Shor)
Constructing quantum computers

Qubits obtained from **physical phenomena**
- Photons (2 polarizations)
- Molecules (2 nuclear spins)
- Superconducting (different)

Major pain: **correction or errors**
- Qubits mixed up with the environment
- Quantum noise
Recent milestone

Partial error correction for a 9-qubit state

Google-sponsored research group
D-Wave

Canadian company, pioneer in QC research

Adiabatic computers, not real QC

512-qubit system

Quantum annealing

No Shor
Many challenges

Stability, error-correction

How much will cost “N quantum operations” vs “N classical operations”? 

Some algorithms need **quantum RAM**, which we don’t really know how to do

Unlikely to come in the next decade, if ever
3. Breaking crypto
TL;DR: We’re doomed

**RSA**: broken

**Diffie-Hellman**: broken

**Elliptic curves**: broken

**El Gamal**: broken
RSA

No more RSA encryption or signatures

Based on the hardness of factoring

You know \( N = p \times q \), you search \( p \) and \( q \)

Hard on a classical computer (most probably)
Easy on a quantum computer!
Shor’s idea to factor $N=pq$

$X^e \mod N$ for $e$ in $[1, 2, 3, ...]$ and some $X$ will repeat with a period dividing $(p-1)(q-1)$

A period gives information on $p$ and $q$!

Shor’s algorithm:
1. Prepare qubits to encode $X, X^2, X^3, X^4, ...$ simultaneously
2. **Find the period** using the Quantum Fourier Transform
3. Exploits the period to **recover p and q**
Discrete logarithms

Problem behind **Diffie-Hellman, ECC**

You know $g$ and $g^y$, you search $y$

Like factoring, a **Hidden Subgroup Problem**

Shor works too!
What about symmetric ciphers?

AES with a 128-bit key:
Classical: 128-bit security
Quantum: 64-bit security

**Grover’s algorithm**: searches in $N$ items in $O(\sqrt{N})$ time and $O(\log N)$ memory

Solution: upgrade to 256-bit AES
4. Post-quantum crypto

hope...
Post-quantum crypto

Alternatives to RSA, Diffie-Hellman, ECC
Resistance to QC can’t be totally proved

http://pqcrypto.org/
Hash-based signatures

Problem: inverting hash functions

Ideas from Lamport (1979), Merkle (1989)

Example of SPHINCS:
(http://sphincs.cr.yp.to/)

- 41 KB signatures
- 1 KB public and private keys
- Slow (100s signatures/sec)
Multivariate signatures

Problem: solve complex systems of equations

First ideas in the 1980s

\[
0 = X_{123}X_3 + X_{13}X_3 + X_2X_4 \\
1 = X_{134}X_4 + X_{23}X_4 \\
0 = X_{13}X_3 + X_{23}X_3
\]

Many schemes have been broken...
Code-based crypto

Problem: decoding error-correcting codes


Limitations:
- Large keys (100 KB+)
- Fewer optimized implementations
Lattice-based crypto

Based on lattice problems (duh!)

**Learning-with-errors**: learn a simple function given results with random noise

Encryption, signature
5. Quantum key distribution
Quantum key distribution (QKD)

Use of quantum phenomena to **share a key**
- Kind of “quantum Diffie-Hellman”
- Not quantum computing
- Not quantum cryptography

“Security based on the laws of physics”
- Eavesdropping will cause errors
- Keys truly random
BB84

First QKD protocol, though not really quantum

Idea:
Send bits in the form of polarized photons
Can be observed in 2 ways, only one is right

<table>
<thead>
<tr>
<th>Alice’s random bit</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice’s random sending basis</td>
<td>+</td>
<td>+</td>
<td>×</td>
<td>+</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>+</td>
</tr>
<tr>
<td>Photon polarization Alice sends</td>
<td>↑</td>
<td>→</td>
<td>↓</td>
<td>↑</td>
<td>↓</td>
<td>→</td>
<td>→</td>
<td>→</td>
</tr>
<tr>
<td>Bob’s random measuring basis</td>
<td>+</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>+</td>
<td>×</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Photon polarization Bob measures</td>
<td>↑</td>
<td>→</td>
<td>↓</td>
<td>→</td>
<td>→</td>
<td>→</td>
<td>→</td>
<td>→</td>
</tr>
<tr>
<td>PUBLIC DISCUSSION OF BASIS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shared secret key</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Caveats

Like any security system, it’s complicated
Security

Eventually relies on **classical crypto**
Typically with frequent rekeying

QKD implementations have been attacked

"Quantum hacking"
(formerly NTNU, Norway)
Deployment

Dedicated optical fiber links

Point-to-point, limited distance (< 100 km)
6. Quantum copy protection
Quantum copy protection

Idea: leverage the no-cloning principle
(cos you can't know everything about something)
Quantum cash

Impossible to counterfeit, *cos' physics* (1969)

Bills include qubits with some secret encoding

Only the bank can authenticate bills...
Publicly verifiable quantum cash

Anyone can verify that a bill isn't counterfeit

Uses public-key crypto, non-quantum

Can be secure even with black-box verification
Quantum software protection

Using quantum techniques:

"Obfuscate" the functionality
Make copies impossible

```plaintext
verify(pwd) {
    return pwd == "p4s5w0rD"
}
```

1. Turn `verify()` into a list of qubits
2. Verification: apply a transform that depends on `pwd`, then measure the qubits
7. Quantum machine learning
Machine learning

“Science of getting computers to act without being explicitly programmed” —Andrew Ng

Successful for spam filtering, fraud detection, OCR, recommendation systems
Machine learning and security

No silver bullet, but may help

ML being used for
  Intrusion detection (network, endpoint)
  Binary vulnerability discovery

Nevertheless, vendors give neither
  Details on the techniques used, nor
  Effectiveness figures or measurements
Quantum machine learning

“Port” of basic ML techniques to QC, like
k-mean clustering
Neural networks
Support vector machines

Many use Grover for a square-root speedup

Potential exponential speedup, but...
Quantum RAM (QRAM)

Awesome concept
Addresses are given in superposition
Read values are retrieved in superposition

Many QML algorithms need QRAM

But it'd be extremely complicated to build
8. Conclusions
Quantum computers s***

Because they...

ARE NOT superfaster computers
WOULD NOT solve NP-hard problems
MAY NEVER BE BUILT anyway
Quantum computers are awesome

Because they...
  Would DESTROY all pubkey crypto deployed
  Give a new meaning to "COMPUTING"
  May teach us a lot about physics and Nature
Thank you!