DIPARTIMENTO
DI MATEMATICA
GIUSEPPE PEANO
UNIVERSITÀ DI TORINO

POLITECNICO
DI TORINO

Dipartimento
di Scienze Matematiche
G.L. Lagrange

The Threat of Quantum Computers to Public-Key Cryptography

Dutto Simone

Group of Cryptography and Number Theory, DISMA, PoliTo (crypto.polito.it)

Math Ph.D. Seminars - May 22nd, 2019

Modern Cryptography

Modern cryptosystems can belong to:

- Symmetric Cryptography:
  the key k is a shared secret among the parties, so that $\mathbf{B} \xrightarrow{E_k(m)} \mathbf{A}$, who evaluates $m = D_k(E_k(m))$.
  Pro: efficient both in hardware and software.
  Con: key distribution through a secure channel.
- Public-Key (or Asymmetric) Cryptography (PKC):
  each party has a public key pk and a secret key sk, then $\mathbf{B} \xrightarrow{E_{pk_\mathbf{A}}(m)} \mathbf{A}$ and $m = D_{sk_\mathbf{A}}(E_{pk_\mathbf{A}}(m))$.
  Pro: no need for a secure channel.
  Con: significantly less efficient.

Public-Key Cryptography

All pre-computers cryptosystems were symmetric (until mid 1970s). When the number of parties started to grow, the requirement of a secure channel became unmanageable.

Thus, PKC was introduced and different cryptosystems, based on various kind of difficult mathematical problems, were adopted. Among the first protocols there were:

- the Diffie-Hellman key exchange (DH, 1976);
- the Rivest-Shamir-Adleman cryptosystem (RSA, 1978).

Diffie-Hellman Key Exchange

The protocol uses the multiplicative group of integers modulo a prime number p, i.e. $\mathbb{Z}_p^\times = \{1, 2, \ldots, p-1\}$:

- **A** and **B** publicly agree to use a modulus p and a generator g of $\mathbb{Z}_p^\times$ (primitive root modulo p);
- **A** chooses $a \in \mathbb{Z}$ and sends $A = g^a \pmod p$ to **B**;
- **B** chooses $b \in \mathbb{Z}$ and sends $B = g^b \pmod p$ to **A**;
- **A** evaluates $s = B^a \pmod p$;
- **B** evaluates $s = A^b \pmod p$.

Now, they share the secret $s = g^{ab} \pmod p$.

The security of DH is assured by the:

Diffie-Hellman Problem (DHP)

Given $p, g \mid \langle g \rangle = \mathbb{Z}_p^\times$, $g^a \pmod p$ and $g^b \pmod p$, what is the value of $g^{ab} \pmod p$?

which is assumed to be hard (Diffie-Hellman assumption) and the most efficient way to solve it is to solve the:

Discrete Logarithm Problem (DLP)

Given p, g as before and $g^x \pmod p$ what is x?

DH is not the only cryptosystem whose security is based on the DLP (DSA, ECC, ElGamal, ...).

Rivest-Shamir-Adleman Cryptosystem

Each user **A** has to follow these steps once:

- choose two prime numbers p and q;
- evaluate $N = p \cdot q$ and $\phi(N) = (p-1)(q-1)$;
- choose $e \,|\, 1 \leq e \leq \phi(N)$, $\gcd(e, \phi(N)) = 1$;
- evaluate $d \,|\, e \cdot d \equiv 1 (\mathrm{mod}\, \phi(N))$;
- $sk_{\mathbf{A}} = (p, q, d)$, $pk_{\mathbf{A}} = (N, e)$.

If **B** wants to send $m \in \mathbb{Z}_N$ to **A**, then he sends $c = E_{pk_{\mathbf{A}}}(m) = m^e (\mathrm{mod}\, N)$. Now **A** can decrypt the received ciphertext using $D_{sk_{\mathbf{A}}}(c) = c^d (\mathrm{mod}\, N)$, since:

$$D_{sk_{\mathbf{A}}}(E_{pk_{\mathbf{A}}}(m)) = m^{e \cdot d} (\mathrm{mod}\, N) =$$
$$= m^{1 + k\phi(N)} (\mathrm{mod}\, N) = m (\mathrm{mod}\, N)$$

The RSA cryptosystem works thanks to the:

Generalized Euler Theorem

Given $N = p \cdot q$, if $a \equiv 1 (\mathrm{mod}\, \phi(N))$ then $\forall m \in \mathbb{Z}$, $m^a \equiv m (\mathrm{mod}\, N)$.

The security of RSA is based on the:

Integer Factorization Problem (IFP)

Given N as before, what are the prime numbers p and q?

In fact, if p and q are known, then $\phi(N)$ can be easily computed and consequently d can be obtained from e.

Other cryptosystems based on the IFP are, for example, Goldwasser-Micali or Rabin.

Classical Algorithms

The strength of the mentioned cryptosystems relies on the assumption that the seen problems are hard to solve.

With standard computers, the solving algorithms have exponential time. In particular, the best algorithm for:

- DLP is the *index calculus*, which has runtime exponential in $p^{1/3}$;
- IFP is the *general number field sieve*, which has runtime exponential in $d^{1/3}$ (d number of digits of N).

Quantum Computers

Quantum computing took form in the 1960s-80s.

In 1981, *Richard P. Feynman* observed the impossibility to simulate a quantum system on a classical computer and proposed a basic model for a quantum computer.

This theoretical innovation brought to new algorithms, like the Shor's algorithm (1994) which allows to solve the DLP and the IFP in polynomial runtime.

The first working quantum computer was built in 2000 and had 5 qubits. Today, they are still large, noisy and unstable, but they reach 50 real qubits (IBM) or over 2000 qubits, but limited to optimization (D-Wave).

Qubits

Qubit (or *quantum bit*) is the basic unit of quantum information: a *two-state quantum-mechanical system*.

The general quantum state of a qubit can be represented by a linear combination of its two orthonormal basis states $|0\rangle$ and $|1\rangle$ (Dirac notation) called *superposition*:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad \text{where } \alpha, \beta \in \mathbb{C},\ |\alpha|^2 + |\beta|^2 = 1\,.$$

The complex values α and β are probability amplitudes related to the basis, i.e., the probability to have 0 as outcome is $|\alpha|^2$ and to have 1 is $|\beta|^2$.

Bloch Sphere

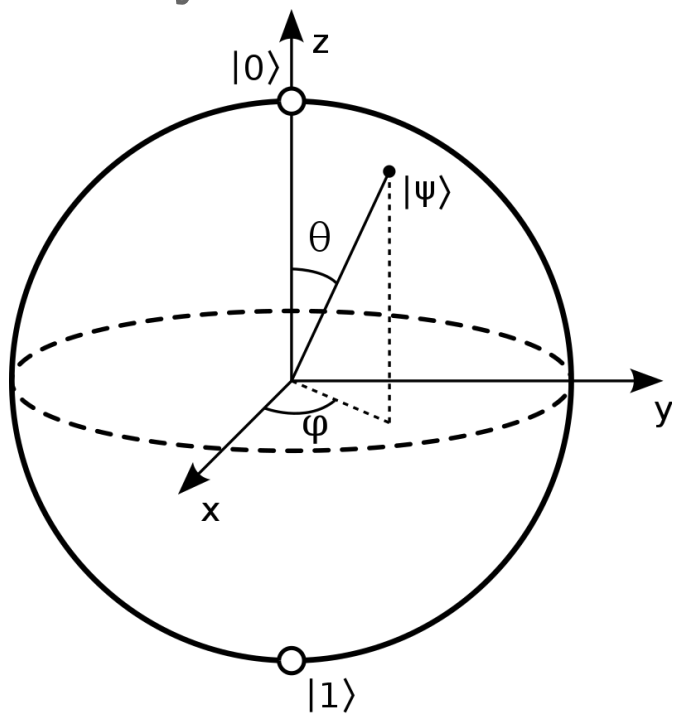
A qubit has 3 degrees of freedom, in coordinates:

$$|\psi\rangle = e^{i\gamma}\left(\cos\left(\frac{\theta}{2}\right)\ |0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)\ |1\rangle\right) .$$

Since the overall phase $e^{i\gamma}$ has no physically observable consequences, α can be arbitrarily chosen to be real ($\gamma = 0$):

$$\alpha = \cos\left(\frac{\theta}{2}\right), \quad \beta = e^{i\varphi}\sin\left(\frac{\theta}{2}\right) .$$

Thus, the possible states of a qubit can be visualized on a sphere called the *Bloch sphere*.



Entanglement

In general, the state of n qubits is described with an amplitude for each possible outcome (n bits).

The state of multiple qubits can be obtained from the state of the single qubits (if they are independent) or not. The second case is called *entanglement*.

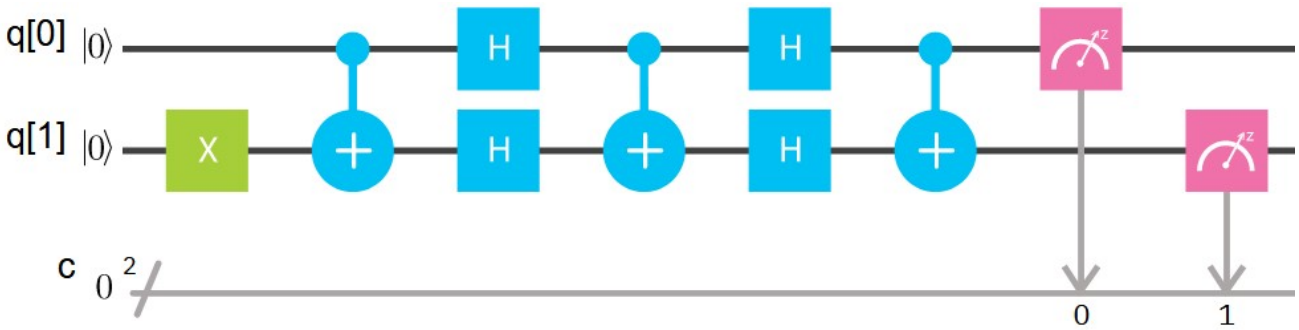
Ex. $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|01\rangle$ can be obtained as combination of the single qubit states $|0\rangle$ and $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$.

$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$ are instead two entangled qubits (a combination of qubits can not be found).

Emulators and GUI

There are different ways to get started with quantum computing: some online tools allow to emulate (Quirk) or also run your scripts on a real quantum computer (IBM Q Experience).

This is an example of the commonly adopted GUI:



The measurement (pink gates) projects the qubit on the z axis (irreversibly) and obtains a bit.

Quantum Logic Gates

In quantum computing, irreversible transformations destroy the quantum nature of qubits.
Thus, only reversible operations are admitted.

If $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, then its vector representation is $\binom{\alpha}{\beta}$ and a gate acting on $|\psi\rangle$ can be represented as a multiplication for the unitary matrix ($U^*U = UU^* = Id$):

$$U = e^{i\phi} \begin{pmatrix} \cos\left(\frac{\rho}{2}\right) - i\sin\left(\frac{\rho}{2}\right) z & -i\sin\left(\frac{\rho}{2}\right)(x - iy) \\ -i\sin\left(\frac{\rho}{2}\right)(x + iy) & \cos\left(\frac{\rho}{2}\right) + i\sin\left(\frac{\rho}{2}\right) z \end{pmatrix},$$

i.e., a rotation of ρ degrees around the axis given by the vector $\hat{n} = (x, y, z)$, multiplied by an overall phase $e^{i\phi}$.

The simplest single-qubit gates are:

- the rotations around the z axis given by:
$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, T = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix},$$
of π, $\frac{\pi}{2}$ and $\frac{\pi}{4}$ degrees respectively;
- the π-rotation around the y axis, $Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$;
- the π-rotation around the x axis, $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, that is the reversible version of a NOT ($|0\rangle \leftrightarrow |1\rangle$);
- the *Hadamard* gate $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$, a π-rotation around $\{x = z, y = 0\}$ which allows to obtain superpositions ($|0\rangle \mapsto \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$).

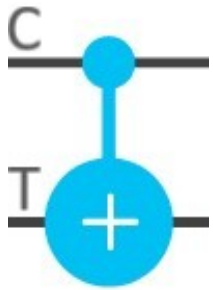
A quantum gate with n inputs is a unitary matrix of size 2^n (the basis contains all combinations of n qubits).

The most interesting and simple multiple-qubits gates are controlled single-qubit gates.

Ex. Controlled NOT:
if C (control) is in state 1, then apply X on T (target). The 2-qubits input is:

$$\alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle,$$

and the unitary matrix multiplying the complex vector $(\alpha, \beta, \gamma, \delta)$ is C-NOT $= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$.



This construction can be generalized to all gates, but can not be used everywhere because of the network topology.

In general, standard multiple-bits gates are not reversible: for some outputs it is impossible to obtain the inputs.

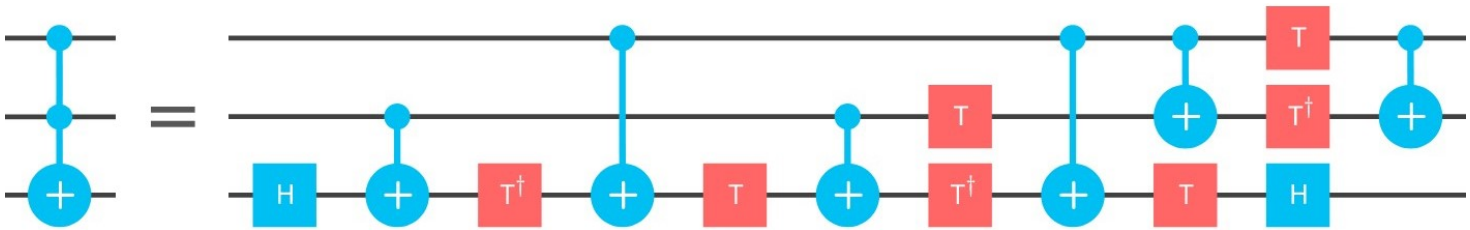
The quantum versions are implemented using more inputs and some controlled gates.

Ex. Quantum AND:

a standard AND is irreversible, because:

$$0 = 0 \,\text{AND}\, 0 = 1 \,\text{AND}\, 0 = 0 \,\text{AND}\, 1\,.$$

The quantum AND exploits the *Toffoli* gate:



where the inputs are the first 2 qubits and the output is the third (only if initialized at $|0\rangle$).

Toy Example

Deutsch-Jozsa problem

A black-box function $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ is either:

- constant ($\forall x \in \mathbb{Z}_2^n \; f(x) = b \in \mathbb{Z}_2$);
- balanced (half inputs are mapped to 0 and half to 1).

Which category does f fall into?

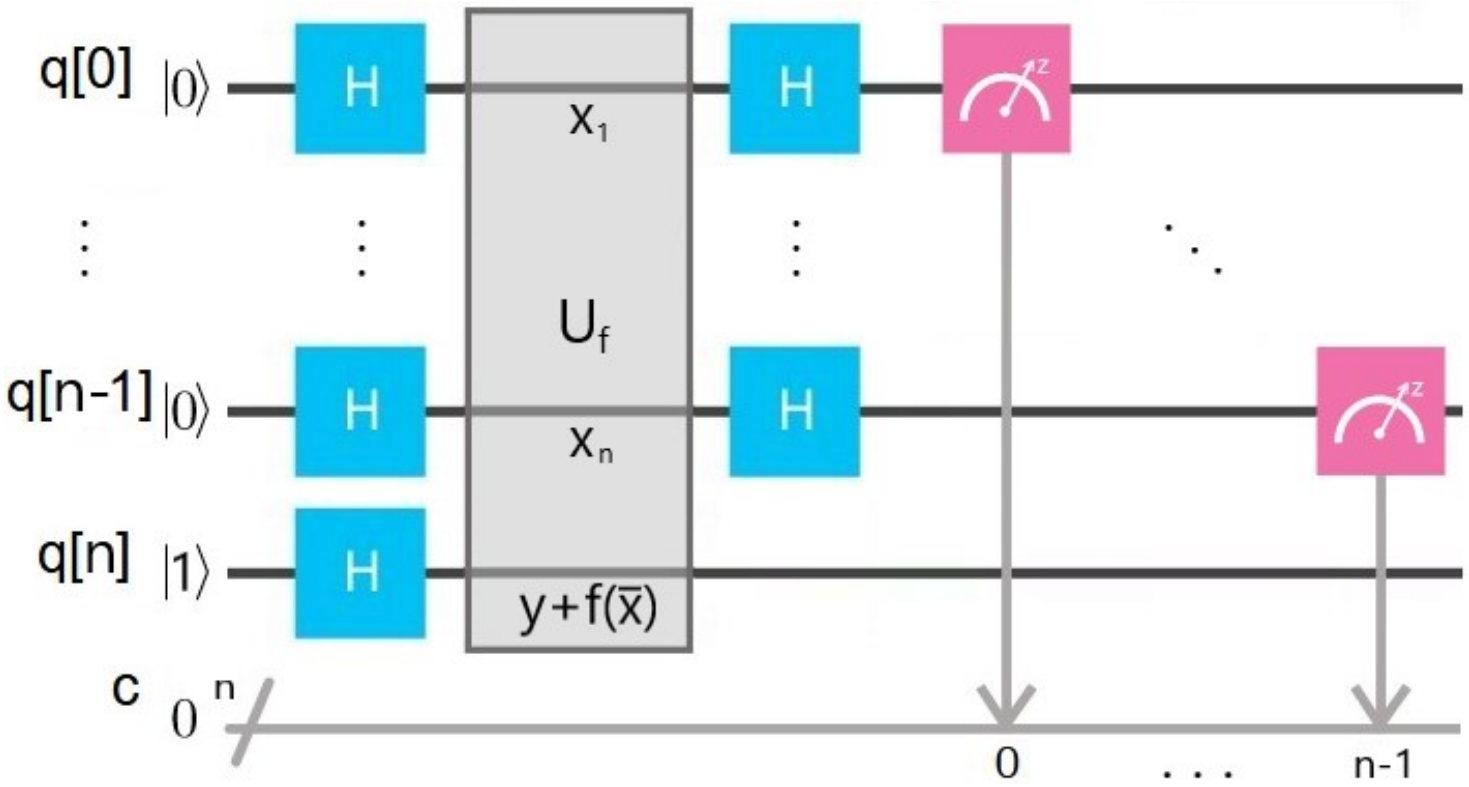
A classical deterministic algorithm requires to evaluate the function f $2^{n-1} + 1$ times in the worst case.

With quantum computing only one evaluation is required.

Let's see the Deutsch-Jozsa algorithm for $n = 1$:

- the initial state is $|01\rangle = |0\rangle|1\rangle$;
- H is applied to both qubits, the obtained state is
$$\tfrac{1}{2}(|0\rangle + |1\rangle)(|0\rangle - |1\rangle)\,;$$
- $U_f : |a\rangle|b\rangle \mapsto |a\rangle|b + f(a)\rangle$ (quantum gate for f) gives
$$\tfrac{1}{2}\Big(|0\rangle\big(|0+f(0)\rangle - |1+f(0)\rangle\big) + |1\rangle\big(|0+f(1)\rangle - |1+f(1)\rangle\big)\Big) =$$
$$= \tfrac{1}{2}\Big(|0\rangle(-1)^{f(0)}\big(|0\rangle - |1\rangle\big) + |1\rangle(-1)^{f(1)}\big(|0\rangle - |1\rangle\big)\Big) =$$
$$= \tfrac{1}{2}\big((-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\big)\big(|0\rangle - |1\rangle\big)\;;$$
- H is applied to the first qubit, whose state becomes
$$\tfrac{1}{2}\Big(\big((-1)^{f(0)} + (-1)^{f(1)}\big)|0\rangle + \big((-1)^{f(0)} - (-1)^{f(1)}\big)|1\rangle\Big),$$
so that, if $f(0) = f(1)$ then only $|0\rangle$ can be evaluated, while otherwise $|1\rangle$ is the only possible state.

For general n, the algorithm is given by:



This algorithm is quite useless, but gives an idea of what quantum computers are capable of.

Problematic Algorithms

The threat of quantum computing to modern cryptography is due to:

- the *Grover*'s algorithm, which allows brute-force with given output using only $O(\sqrt{N})$ evaluations of the function (N size of the domain). This quadratic speedup bothers also symmetric cryptography, but it is sufficient to double the size of the keys;
- the *Shor*'s algorithm, a period-finding quantum routine which allows to theoretically break all PKC based on the IFP or on the DLP.

Shor's Algorithm

How to break the IFP

Suppose $N = p \cdot q$ with d decimal digits.

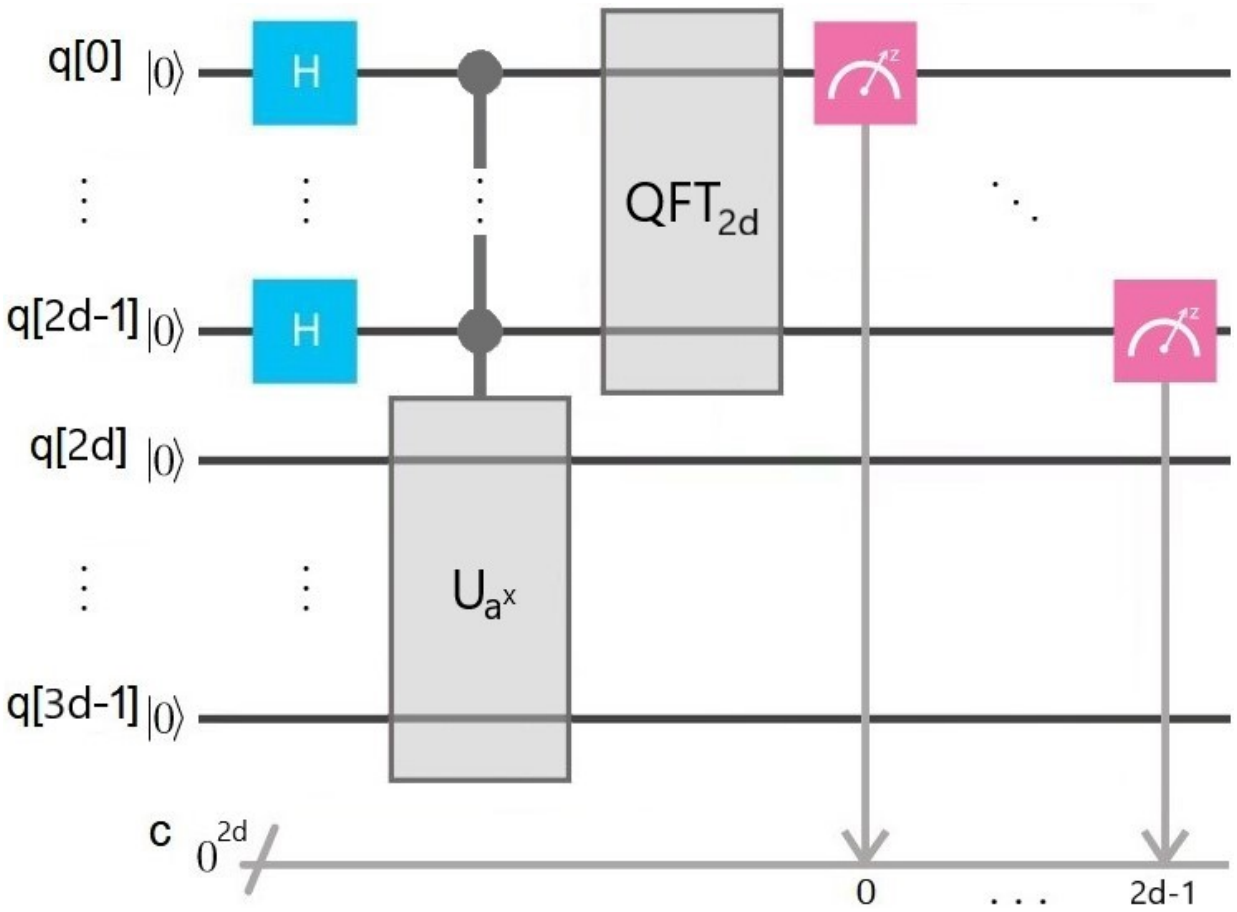
Shor's algorithm can factorize N in a runtime polynomial in d. The procedure consists of two parts:

- a classical algorithm to reduce the IFP to the Period-Finding Problem for $f(x) = a^x \pmod N$ and use the resulting period r to factorize N;
- a quantum algorithm to solve the PFP.

The classical part consists in:

1. pick $a < N$;
2. compute $\gcd(a, N)$ (*Euclidean* algorithm);
3. if $\gcd(a, N) \neq 1$, then $p = \gcd(a, N)$, stop;
4. else, find the period r of a (quantum part);
5. if r is odd (low probability), then return to 1.;
6. else, $N|(a^r - 1) = (a^{\frac{r}{2}} - 1)(a^{\frac{r}{2}} + 1)$, where $N \not| (a^{\frac{r}{2}} - 1)$ (otherwise $r = \frac{r}{2}$);
7. if $N|(a^{\frac{r}{2}} + 1)$, then return to 1.;
8. else, $p = \gcd(a^{\frac{r}{2}} - 1, N)$ and $q = \gcd(a^{\frac{r}{2}} + 1, N)$, stop.

The quantum part is depicted in the figure.

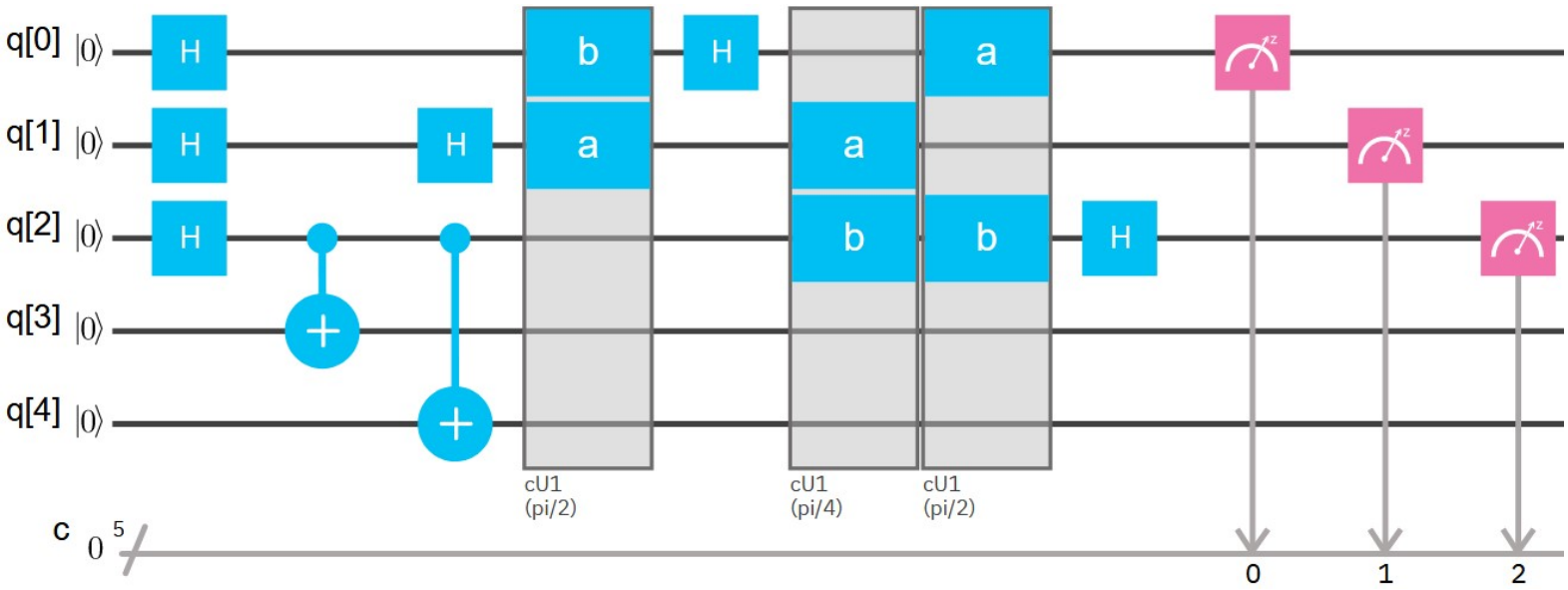


The measured output is a multiple of $\frac{2^{2d}}{r}$ and can be used to find the order r of a.

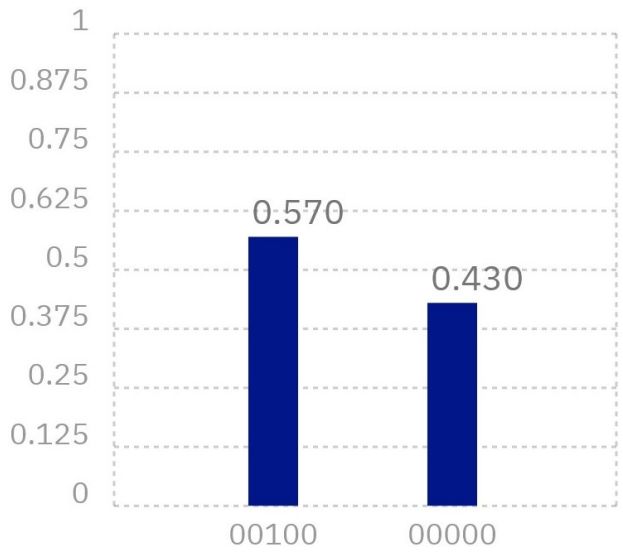
Example

Let's solve the IFP for $N = 15$.

- $a = 11, \gcd(a, N) = 1$ so let's find the period r of a;
- since $15 < 16 = 2^4 = 2^d$, the quantum subroutine requires $3d = 12$ qubits (and 196 gates). This compiled version uses only 5 qubits (and 11 gates):



- the measured values, i.e. the multiples of $\frac{2^3}{r}$, are:



  where the only acceptable result is $(00100)_2 = 4$. Thus, the period of $a = 11$ is $r = \frac{8}{4} = 2$;
- then $15 \mid (11^2 - 1) = (11 - 1)(11 + 1) = 10 \cdot 12$. Since $\gcd(10, 15) = 5$ and $\gcd(12, 15) = 3$ the resulting factorization is $15 = 5 \cdot 3$.

Shor's Algorithm

How to break the DLP

Suppose $y = g^x \pmod p$ with $\langle g \rangle = \mathbb{Z}_p^\times$.

Shor's algorithm can find x in a runtime polynomial in p. Again the QFT is exploited to solve the PFP for a particular function: $f(x_1, x_2) = g^{x_1} y^{x_2}$.

The resulting period is a pair (r_1, r_2) such that
$f(x_1 + r_1, x_2 + r_2) = f(x_1, x_2) \Leftrightarrow g^{r_1} y^{r_2} \equiv 1 \pmod p \Leftarrow$
$\Rightarrow g^{r_1 + x r_2} \equiv 1 \pmod p \Leftrightarrow r_1 + x r_2 \equiv 0 \pmod{p-1} \Leftarrow$
$\Rightarrow x \equiv -\frac{r_1}{r_2} \pmod{p-1}$.

All is not lost...

In December 2016 the NIST (National Institute of Standards and Technologies) opened a call for quantum-resistant PKC proposals (NISTIR 8105).

This first round counted 69 submissions and in January 2019, after 2 years of cryptanalysis performed by the scientific community, only 26 proposals were selected for the second round (NISTIR 8240).

The idea is to obtain by 2025 some quantum-resistant recognized PKC algorithms.

Thank you for your attention.

simone.dutto@polito.it

https://crypto.polito.it