The Quantum Algorithm for Factoring

John Livieratos

January 08 2015

MPLA, Algorithms and Complexity 1.

Introduction

Our aim is to present an efficient algorithm for FACTORING. This algorithm was first discovered by Peter Shor in 1994. The catch is that this algorithm needs a quantum computer to be implemented and there are questions regarding the possibility that such a computer can be build. We save this discussion for the end of the presentation.

Qubits, Superposition, Amplitude

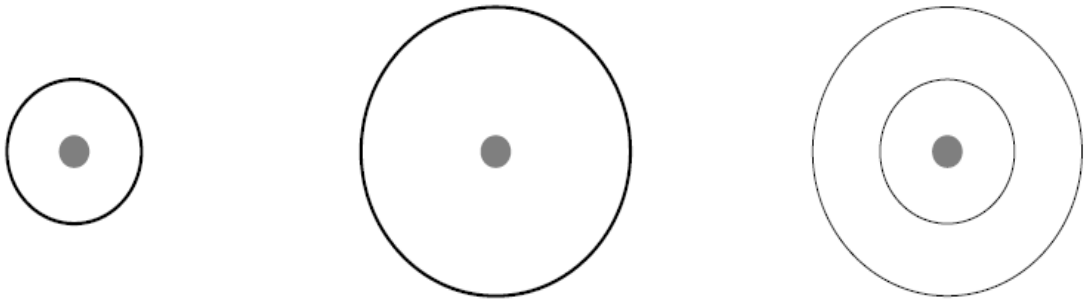
We begin by some basic features from quantum physics that are necessary to understand the algorithm and quantum computers in general.

- Let $|0\rangle$ be the ground state (lowest energy configuration) of a single electron in the hydrogen atom, and $|1\rangle$ the excited state (high energy configuration). In classical physics, these are the two possible states of the electron.
- The superposition principle: If a quantum system can be in one of two states, it can also be in any *linear superposition* of these two states. That is

$$a_0|0\rangle + a_1|1\rangle, a_0, a_1 \in \mathbb{C}, |a_0|^2 + |a_1|^2 = 1$$

- a_0, a_1 are called the *amplitudes* of their respective states, and the above superposition is the basic unit of encoded information in quantum computers, called *qubit* .

Superposition



ground state $|0\rangle$ excited state $|1\rangle$ superposition $\alpha_0|0\rangle + \alpha_1|1\rangle$

Measurement

The electron's superposition is its private world. To find the electron's state, we must make a *measurement*. The result of a measurement is always a single bit, 0 or 1.

If the electron's superposition is $a_0|0\rangle + a_1|1\rangle$, then the outcome of the measurement is 0 with probability $|a_0|^2$ and 1 with probability $|a_1|^2$.

Another difference from the classical physics, is that the act of measurement causes the system to change it's state. That is, if the outcome of the measurement is 0, then the new state of the system is $|0\rangle$, and if it's 1, the new state is $|1\rangle$.

k-LEVEL SYSTEMS, $k \geq 2$

In reality, a system (the electron of the hydrogen atom for example) can have many energy levels (states): the ground state, the first excited state, the second excited state and so on.
In a k-level system, we would denote these states as $|0\rangle, |1\rangle, |2\rangle, ..., |k-1\rangle$.
By the superposition principle, we have the general quantum state of the system as follows:

$$a_0|0\rangle + a_1|1\rangle + ... + a_{k-1}|k-1\rangle$$

where $\sum_{j=0}^{k-1} |a_j|^2 = 1$

As before, a measurement will disturb the system, and resulting in a number $j \in \{0, ..., k-1\}$ (with probability $|a_j|^2$) will force the system to enter the state $|j\rangle$.

Partial measurement and Entanglement

We now consider the case of two qubits in a 2-level system.

- In *classical physics* we have four possible states: $00, 01, 10, 11$
- In *quantum physics*, due to the superposition principle, we have

$$|\alpha\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle$$

  where $\sum_{x\in\{0,1\}^2} |a_x|^2 = 1$
- *Measuring* the system now will reveal two bits of information $i, j \in \{0, 1\}$ with probability $|a_{ij}|^2$ and the system will enter the state $|ij\rangle$, which means that the first electron is in the i-th excited state and the second in the j-th (ground state if i and/or j are equal to 0).

Partial measurement and Entanglement

We examine, with an example, what happens in the case of a partial measurement:

- $Pr\{1st\ bit = 0\} = Pr\{00\} + Pr\{01\} = |a_{00}|^2 + |a_{01}|^2$
- We obtain the new *superposition* of the system firstly by crossing all terms of $|\alpha\rangle$ that are inconsistent with the result of the partial measurement (those whose first digit is 1) and then *renormalizing*, since the sum of the squares of the remaining amplitudes is no longer 1:

$$|\alpha_{new}\rangle = \frac{a_{00}}{\sqrt{|a_{00}|^2 + |a_{01}|^2}}|00\rangle + \frac{a_{01}}{\sqrt{|a_{00}|^2 + |a_{01}|^2}}|01\rangle$$

- Finally, we notice that given an arbitrary state of two (or more) qubits, it's not possible to specify the state of each individual qubit. We consider for example one of the *Bell states*: $|\psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$. This is another major difference from the classical physics, called *entanglement*.

Quantum Algorithm

Let's examine now the function of a quantum algorithm.

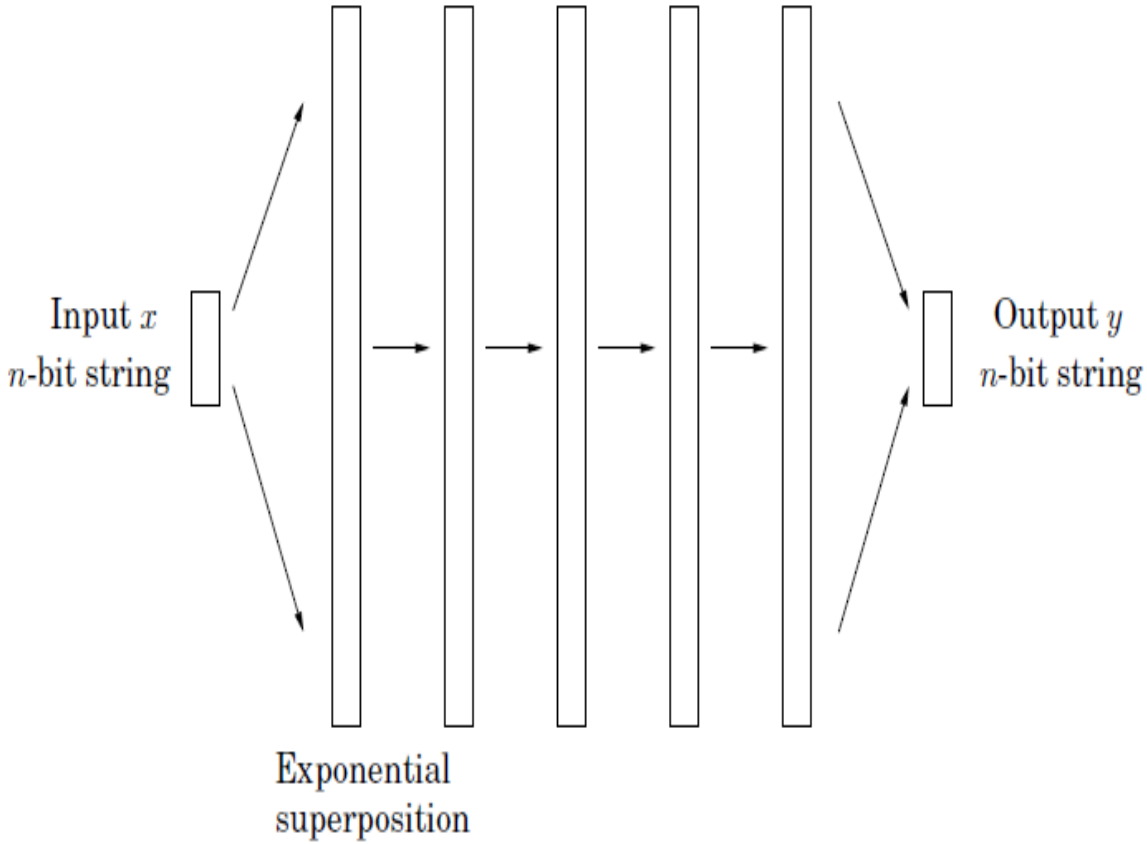
Let the *input* of the algorithm be an n-bit string x. Thus, the algorithm takes as input n qubits in state $|x\rangle$. Now for the quantum stage of the algorithm, it is helpful to think of it as having two stages.

At the first stage, the n qubits form an exponentially large superposition, which is set up to have some underlying *pattern* or *regularity* that, when detected, will solve the task at hand. The state of the qubits now is some superposition $\sum_y \alpha_y |y\rangle$

At the second stage, a series of quantum operations is applied to the input, revealing the pattern.

Finally, a measurement is performed, resulting in an output of an n-bit string y with probability $|a_y|^2$. The randomness of the output is of small concern, if the probability that it is right is high enough.

Quantum Algorithm



Elementary Quantum Gates and Circuits

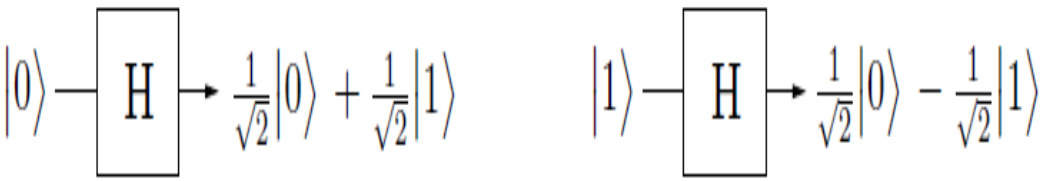
We will now have a brief discussion about some basic quantum gates and circuits.

- Hadamard Gate:

  $H\left(|0\rangle\right) = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ & $H\left(|1\rangle\right) = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$

  We notice that measuring the resulting qubit in either case, results in 0 or 1 with probability $\frac{1}{2}$. Finally, for input an arbitrary superposition of a qubit:

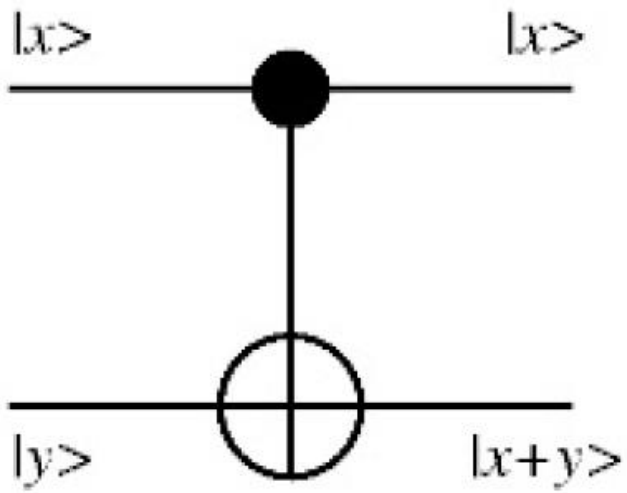
$$H\left(a_0|0\rangle + a_1|1\rangle\right) = a_0 H\left(|0\rangle\right) + a_1 H\left(|1\rangle\right) = \frac{a_0 + a_1}{\sqrt{2}}|0\rangle + \frac{a_0 - a_1}{\sqrt{2}}|1\rangle$$



Elementary Quantum Gates

- Controlled-NOT or CNOT gate: This gate operates upon two qubits. The first acts as the control qubit and the second as the target. This gate flips the second bit iff the first qubit is 1:

$$CNOT\left(|00\rangle\right) = |00\rangle$$
$$CNOT\left(|01\rangle\right) = |01\rangle$$
$$CNOT\left(|10\rangle\right) = |11\rangle$$
$$CNOT\left(|11\rangle\right) = |10\rangle$$



Elementary Quantum Gates

Let $|\alpha\rangle = \sum_{x\in\{0,1\}^n} \alpha_x|x\rangle$ be an arbitrary quantum state on n qubits. The result of applying the Hadamard Gate to the first qubit is:

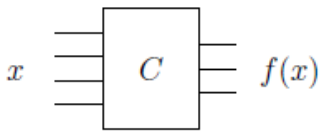
$$|\beta\rangle = \sum_{x\in\{0,1\}^n} \beta_x|x\rangle \text{ where } \beta_{0y} = \frac{\alpha_{0y} + \alpha_{1y}}{\sqrt{2}} \ \& \ \beta_{1y} = \frac{\alpha_{0y} - \alpha_{1y}}{\sqrt{2}}$$

So, by applying the Hadamard gate to only one of the n qubits, changes all the 2^n amplitudes.
This is what will give us an exponential speedup to the following algorithms.

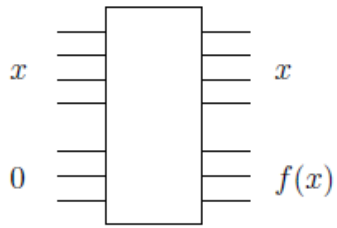
Basic Quantum Circuits

Lastly, we briefly explore two quantum circuits:

- *Quantum Fourier Transform:* These circuits take as input n qubits in some state $|\alpha\rangle$ and, by applying the QFT, output the state $|\beta\rangle$. We will not go in any more detail about these circuits, but we will discuss later the QFT algorithm.
- *Classical Functions:* Let f be a function with n input bits and m output bits. Suppose that there exists a classical circuit that outputs $f(x)$. Then, there exists a quantum circuit that, on input the n-bit string x padded with m 0's, outputs $x \,\&\, f(x)$. The input of this circuit can be a superposition over the n-bit strings x: $\sum_x |x, 0^k\rangle$. Then, the output would be $\sum_x |x, f(x)\rangle$.



Classical circuit



Quantum circuit

QFT

We will now briefly discuss the quantum version of the FFT algorithm. The FFT algorithm takes as input an M-dimensional complex-valued vector α (with $M = 2^m$), and outputs an M-dimensional complex valued vector β.
In the following figure, ω is a complex Mth root of unity and the $\sqrt{M}$ is an extra factor to ensure that:
$\sum_{i=0}^{M-1} |\alpha_i|^2 = 1 \Rightarrow \sum_{i=0}^{M-1} |\beta_i|^2 = 1$.

$$\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{M-1} \end{bmatrix} = \frac{1}{\sqrt{M}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{M-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(M-1)} \\ & \vdots & & & \\ 1 & \omega^j & \omega^{2j} & \cdots & \omega^{(M-1)j} \\ & \vdots & & & \\ 1 & \omega^{(M-1)} & \omega^{2(M-1)} & \cdots & \omega^{(M-1)(M-1)} \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{M-1} \end{bmatrix}$$

QFT

Recall that the FFT can perform this calculation in $O(MlogM)$ steps. The QFT can do the same calculation in $O\left(log^2 M\right)$ steps.

- *Input:* A superposition of $m = logM$ qubits, $|\alpha\rangle = \sum_{i=0}^{M-1} \alpha_i |i\rangle$
- Using $O\left(log^2 M\right)$ quantum operations, QFT transforms the superposition in a way that at each stage, the superposition encodes the intermediate results of the corresponding stage of the classical FFT. The result is the superposition $|\beta\rangle = \sum_{i=0}^{M-1} \beta_i |i\rangle$. We will avoid to get into any further details about the exact procedure followed by the QFT.
- *Output:* A random $logM$-bit number $0 \leq i \leq M-1$ with probability $|\beta_j|^2$, that is the product of measurement.
- There is a very important difference between the FFT and the QFT. The first actually outputs the $\beta_0, ..., \beta_{M-1} \in \mathbb{C}$, where the second prepares a superposition $|\beta\rangle = \sum_{i=0}^{M-1} \beta_i |i\rangle$ and ouputs just an index $|i\rangle$. It turns out that there are applications that this is all we need to get by QFT.

Periodicity

It is evident that we can see $|\alpha\rangle$ as the vector $(\alpha_0, ..., \alpha_{M-1})$.

Define $|\alpha\rangle$ as *periodic* with *period k* and *offset l* if $a_i = a_j$ whenever $i \equiv j \ mod \ k$ - that is, $|\alpha\rangle$ consists of $\frac{M}{k}$ repetitions of the same sequence $(\alpha_0, ..., \alpha_{k-1})$ - and exactly on of the k numbers $\alpha_0, ..., \alpha_{k-1}$, say a_l, is nonzero.

Now, if the input is periodic, we can use QFT -or quantum Fourier sampling as is its other common name- to compute its period. This is due to the fact that, if the input is periodic with period k, for some k that divides M, the output will be any one of the k multiples of $\frac{M}{k}$ with equal probability. We again omit the proof of this claim.
Now, repeating the sampling a few times and taking the gcd of the results will, with high probability, give us the number $\frac{M}{k}$. From that we can get the period k.

FACTORING

At last, we can begin discussing our target, the efficient algorithm for FACTORING.
The algorithm to factor an integer N is, in essence, a series of reductions. We begin by reducing FACTORING to finding a nontrivial square root of $1\ modulo\ N$:

- For any integer N, a *nontrivial square root of* 1 *modulo* N is any integer $x \neq \pm 1\ mod\ N$ such that $x^2 \equiv 1\ mod\ N$.
- *Lemma1:* x nontrivial square root of $1\ mod\ N \Rightarrow gcd\,(x+1, N)$ is a nontrivial factor of N.

Proof.

$x^2 \equiv 1\ mod\ N \Rightarrow N | \left(x^2 - 1\right) = (x-1)(x+1)$. But $x \neq \pm 1\ mod\ N$, so N doesn't divide $(x-1)$ or $(x+1)$. Thus, N has a nontrivial common factor with $x+1$. Therefore, $gcd\,(N, x+1)$ is a nontrivial factor of N.

FACTORING

We now reduce finding a nontrivial square root of $1\ mod\ N$ to computing the order of an integer.

- The *order* of $x\ mod\ N$ is $min\{r \in \mathbb{Z}^+ |\ x^r \equiv 1\ mod\ N\}$.
- *Lemma2:* Let N be an odd composite number, with at least two distinct prime factors. Let x be chosen uniformly at random from $\{0, ..., N-1\}$. If $gcd\,(x, N) = 1$, then, with probability at least $\frac{1}{2}$, the order r of $x\ mod\ N$ is even and $x^{\frac{r}{2}}$ is a nontrivial square root of $1\ mod\ N$.
- From Lemma1 and 2, we have found, with good probability, a nontrivial factor of N, namelly $gcd\left(x^{\frac{r}{2}} + 1, N\right)$, effectively reducing FACTORING to ORDER FINDING.
- Now, ORDER FINDING has a periodic function associated with it. For fixed N and x, we consider the function $f\,(\alpha) = x^{\alpha}\ mod\ N$. If r is the order of x, then $f\,(kr) = 1\ \&\ f\,(kr+1) = x\ \forall k \in \mathbb{N}$. Thus f is periodic with period r, and we can compute it efficiently.

Creating a periodic superposition

We aim now to set up a periodic superposition using the function $f(\alpha) = x^{\alpha} \ mod \ N$.

- We start with two quantum registers, initially 0.
- We compute the QFT of the first register modulo M, to get a superposition over all numbers in $\{0, ..., M-1\} : \frac{1}{\sqrt{M}} \sum_{\alpha=0}^{M-1} |\alpha, 0\rangle$. We can show that if the initial superposition has period k, the new has period $\frac{M}{k}$. Here the initial had period M, so the new will have period 1.
- We now compute the function $f(\alpha) = x^{\alpha} \ mod \ N$. The quantum circuit regards the contents of the first register as input, and so it outputs, at the second register, $\frac{1}{\sqrt{M}} \sum_{\alpha=0}^{M-1} |\alpha, f(\alpha)\rangle$.
- Finally, we measure the second register. This gives a periodic superposition on the first register, with period r, the period of f:

Proof

- Since f is periodic with period r, for every rth value in the first register, the contents of the second register are the same.
- The measurement therefore gives us $f(k)$ for some random $k \in \{0, ...r-1\}$.
- Due to the partial measurement, the first register is now in a superposition of only those values in α that are compatible with the measurement.
- But these values are exactly $k, k+r, ..., k+M-r$. So the resulting state of the first register is a periodic superposition $|\alpha\rangle$ with period r, the order of x we wanted to find.

FACTORING

Let's put together all the pieces of the FACTORING quantum algorithm that we've discussed.
Since we now that we can efficiently check whether the input is prime, we suppose the input is an odd composite number, with at least two distinct prime factors.

- Choose x uniformly at random from $\{1, ..., N-1\}$
- Let M be a power of 2 near N ($M \approx N^2$, for reasons we will not discuss)
- Repeat $s = 2logN$ times: Do the process described previously and then Fourier sample the superposition $|\alpha\rangle$ to obtain an index $j_i \in \{0, ..., M-1\}$
- Set $g = gcd\,(j_1, ..., j_s)$
- If $\frac{M}{g}$ is even, compute $gcd\left(N, x^{\frac{M}{2g}} + 1\right)$ and output it if its a nontrivial factor of N, otherwise redo the whole process.
- FACTORING is now solved in $O\left(n^3 logn\right)$ steps.

A short discussion

- If we manage to build quantum computers, then, due to the quantum factoring algorithm, the systems that are based in RSA cryptosystem will no longer be secure.
- Quantum computers violate the *extended Church-Turing Thesis*, that all ways of implementing computers are polynomially equivalent.
- So far, the most ambitious quantum computation was the factorization of 15 into 3 and 4 using nuclear magnetic resonance, but there are questions concerning the implementation of the quantum factoring algorithm.
- If quantum computers cannot be build, maybe this points to a fundamental flaw in quantum physics.

" I think I can safely say that no one understands quantum physics "

-Richard Feynman, $1918 - 1988$