Quantum algorithms

Lecture 3: Quantum algorithms (2)

Benoît Vermersch

October 9, 2023

LPMMC Grenoble

Outline

Exponential speedup: Shor's algorithm

- Reduction to order finding
- Order finding quantum subroutine

Outline

Exponential speedup: Shor's algorithm

- Reduction to order finding
- Order finding quantum subroutine

Shor's algorithm

- Problem: Given N, find non-trivial factors $N = pq$.
- Complexity: Best known algorithm is sub-exponential in the number of digits $\sim \log(N)$.
- Shor's algorithm with polynomial complexity in $\log(N)$ offers an exponential speedup.

Shor's algorithm: Number theory

- For a given $1 < a < N$, a coprime with N, we introduce the order r, as the smallest strictly positive integer such that

$$a^r = 1 \bmod(N)$$

Additional useful facts on the order r

- Existence:
  - Define $f(x) = a^x \mod(N)$
  - Let us denote a doublon (s, t) $f(t) = f(s)$ (It exists since the output space is finite)
  - a is coprime with N, therefore there exists an integer a^{-1} such that $aa^{-1} = 1 \mod(N)$ (Bezout's theorem)
  - This implies $a^{t-s} = 1 \mod(N)$. Therefore r exists.
- In addition,
  - We also have f takes different values in $[1, r-1]$.
  - If this were not true, the previous derivation would show that r is not the minimal number achieving $a^r = 1 \mod(N)$.

Shor's algorithm: Number theory

- Theorem: If r is even, let us define $b = a^{r/2}$. If, in addition, $b \neq -1 \bmod(N)$, then

$$p = gcd(b-1, N) \text{ and } q = gcd(b+1, N) \text{ are non-trivial factors of } N$$

Shor's algorithm: Number theory

- Proof: For, e.g, $p = gcd(b-1, N)$:
    - If $p = N$, N divides $b-1$, therefore $a^{r/2} = 1 \mod(N)$, which contradicts the fact that r is the order of a.
    - If $p = 1$, there are integers (u, v) such that (Bézout's theorem)

$$(b-1)u + Nv = 1 \implies (b^2-1)u + N(b+1)v = b+1 \tag{1}$$

    - This implies N divides $b+1$, which implies $b = -1 \mod(N)$, another contradiction.

Shor's algorithm

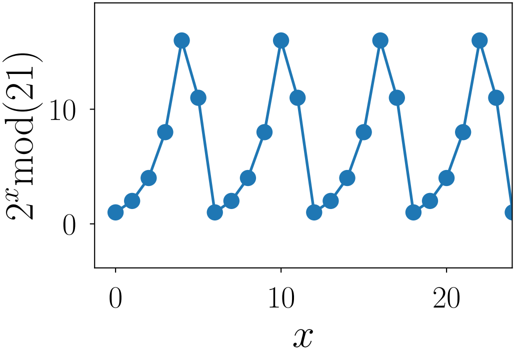
The algorithm (1994)

1. Pick $1 < a < N$ random, and check if a is coprime with N.
2. Find order r via quantum subroutine
3. If r is even, let us define $b = a^{r/2}$. If, in addition, $b \neq -1 \mod(N)$, then $p = gcd(b-1, N)$ and $q = gcd(b+1, N)$ are non-trivial factors of N.
4. Otherwise, go back to step 1.

- Note: The *gcd* operation can be performed efficiently on a classical computer.
- Existence and 'likelihood' conditions of such even r with $b \neq -1 \mod(N)$: Beyond the scope of this course $\rightarrow$ c.f., J. Preskill's lectures

Shor's algorithm

- Example $N = 21$ (see TD2)
- We pick $a = 2$, we find $r = 6$, as $a^6 = 1 \bmod(N)$.



Shor's algorithm

- Example $N = 21$ (see TD2)
- We pick $a = 2$, we find $r = 6$, as $a^6 = 1 \mod(N)$.
- We have that r is even, we define $b = a^{r/2} = 8$
- We have that $b \neq -1 \mod(21)$.
- Thus, we find that $gcd(21, 7) = 7$ divides N
- What about $N = 14351$?

Shor's algorithm

```
from pylab import *
N = 14351
for i in range(10):
    a = randint(1,N)
    print('a=',a)
    g = gcd(a,N)
    print('gcd(a,N)=',g)
    if g==1:
        r = 1
        ### I simulate here the quantum subroutine with an exponentially costly for
        while (r<=N):
            if (a**r)%N==1:  #Check if r is the order of (a,N)
                if r%2==0:
                    print('r ', r)
                    b = a**(r//2)
                    print('b ',b)
                    if (b+1)%N>0: print(gcd(b-1,N), ' divides ',N)
                    else: print('fail')
                else: print('fail')
                print('——')
```

Reduction of order finding to period finding

- Notice that $f(x) = a^x \mod(N)$ is periodic in r.
- Proof: We have that

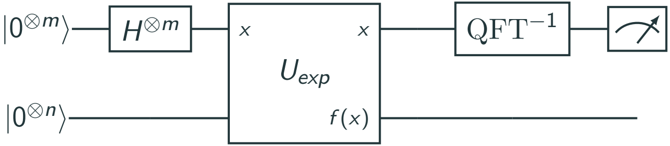
$$f(x+r) = a^{x+r} \mod(N) = a^x(1+Nt') \mod(N) = f(x) \tag{2}$$

Therefore f is periodic in r

- Also, because f takes different values in $[1, r-1]$, we have that $f(x_1) = f(x_2)$ iff $x_1 = x_2 \mod (r)$.
- We can find this period up to excellent approximation using the Quantum Fourier Transformation (QFT) operation.

Order finding quantum subroutine

- Classical input: the function $f(x) = a^x \mod(N)$.
- Classical ouput: the period r.



- To provide enough 'spectral resolution', i.e., represent sufficiently large numbers x, we choose m, such that $M = 2^m > N^2$. We have that $2^n > N$ for the second register.

Order finding quantum subroutine

- The first steps, *modular exponentiation*, create the following state with order $O(m^3)$ gates (can be improved for large m, see Preskill notes) 'We load the entire function in the Hilbert space via quantum parallelism'

$$|\psi\rangle = \frac{1}{\sqrt{M}} \sum_x |x\rangle \otimes |f(x)\rangle$$

- Note: As usual, we make here the correspondence $|x\rangle = |x_1, \ldots, x_m\rangle$ to encode an integer x in a qubit register. Same thing for $f(x)$ in the second qubit register of size n.

Order finding quantum subroutine

- The second step is the inverse quantum Fourier Transform, realizable with $O(m^2)$ gates (see TD2), with unitary circuit

$$\mathrm{QFT}^{-1}|x\rangle = \frac{1}{\sqrt{M}} \sum_y e^{-2i\pi xy/M} |y\rangle$$

Order finding quantum subroutine

- Before the measurement, the quantum state reads

$$|\psi\rangle = \frac{1}{M} \sum_{x,y} e^{-2i\pi xy/M} |y, f(x)\rangle \tag{3}$$

- The probability to measure the bitstring y is

$$P(y) = \langle\psi|(|y\rangle\langle y| \otimes \mathbf{1})|\psi\rangle = \frac{1}{M^2} \sum_{x_1,x_2} e^{2i\pi y(x_1 - x_2)/M} \langle f(x_1)|f(x_2)\rangle \tag{4}$$

- Now we can use $f(x_1) = f(x_2)$ iff $x_1 = x_2 \mod(r)$...

Order finding quantum subroutine

- The probability to measure the bitstring y is

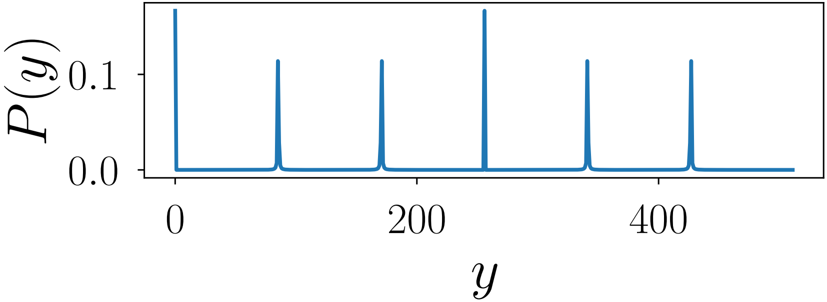
$$P(y) = \frac{1}{M^2} \sum_t \alpha_t e^{2i\pi yrt/M} \tag{5}$$

with $\alpha_t = \#[(x_1, x_2) | x_1 - x_2 = rt]$.

- $P(y)$ is maximal when $ry/M \approx p$, p integer, i.e when $y/M \approx p/r$
- The peaks $\tilde{y}$ in $P(y)$ can be used to extract $r \approx p\tilde{y}/M$ with high success probability (for a sufficiently large value of M, using the continous fraction algorithm)
- Large M offers good 'spectral resolution'. That's why we choose $M \geq N^2$.

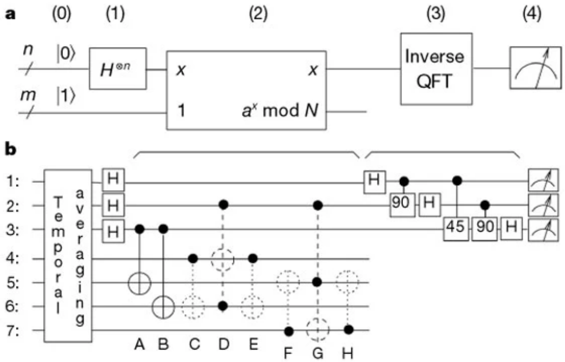
Order finding quantum subroutine

- Illustration with $N = 21$, $M = 512$, $a = 2$, $r = 6$ [Exercices 3]



Experimental realization for $N = 15$

- Vandersypen et al, Nature 2001



- Important technological achievement
- What limits the current applications to small numbers?
  $\rightarrow$ Lecture 4: Quantum error correction