CS290A, Spring 2005:

Quantum Information & Quantum Computation

Wim van Dam

Engineering 1, Room 5109
vandam@cs

http://www.cs.ucsb.edu/~vandam/teaching/CS290/

Administrivia

- Thursday, May 12: Talk by M. Steffen on “Nuclear Magnetic Resonance” (NMR) quantum computing.
- Handout will contain explanation of an efficient implementation of the quantum Fourier transform.
- Again, Final will be an exam *à la* last week’s Midterm
- Questions?

Recapitulation

- There is no straightforward quantum algorithm to solve NP-complete problems ($\Theta(\sqrt{N})$ bound on searching).
- We have to look at problems that —we think—are not in P (classically) but not NP-complete either.
- [Shor'94] Quantum computers can efficiently solve Factoring and Discrete Logarithms. This is done by the quantum algorithm for **period finding** (using the **quantum Fourier transform**).

Quantum Fourier Transform

Consider the mod N numbers {0,1,2,…,N–1}.
The "Quantum Fourier Transform over $\mathbb{Z}_N$" is defined for each $x \in \{0,1,\ldots,N-1\}$ by

$$\boxed{|x\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i \cdot xy/N} |y\rangle}$$

Hence for each superposition over mod N:

$$\boxed{\sum_{x=0}^{N-1} \alpha_x |x\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \sum_{x=0}^{N-1} \alpha_x \cdot e^{2\pi i \cdot xy/N} |y\rangle}$$

Important fact: The QFT can be efficiently implemented in circuit size poly(log(N)) for each N.

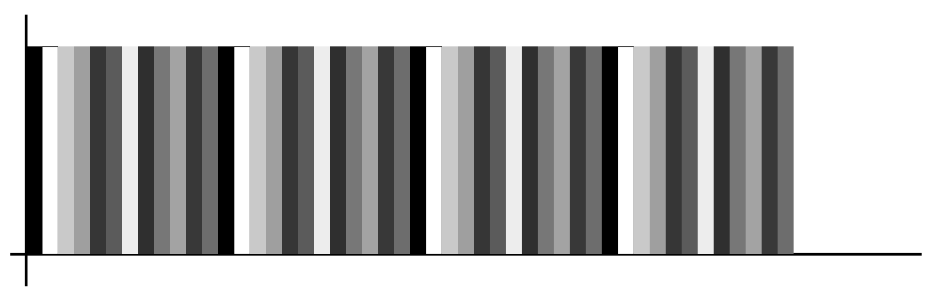
Periodicity Problem

Consider function $F:\{0,\ldots,N-1\} \rightarrow S$

<u>Assume that</u>: F has period r
F is bijective on its period

$$\boxed{F(x) = F(y) \text{ if and only if } x = y \bmod r}$$

<u>Task</u>: determine r (efficiently ~ poly(log N))



0………………………………N
0……...r…….2r…….3r……4r

$F:\{0,1\}\rightarrow\{$  $\}$

Periodicity Algorithm

1) Create superposition of F(x) values: $\boxed{\frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x,F(x)\rangle}$

2) Measure the rightmost F-register. This will give a random value F(c), and because of the periodicity F(c) = F(c+r) = F(c+2r) = …the left state is now: $\boxed{\sqrt{\frac{r}{N}}\sum_{t=0}^{N/r-1}|tr+c\rangle}$

3) Apply the Fourier transform over {0,1,…,N–1} , yielding

$$\boxed{\frac{\sqrt{r}}{N}\sum_{j=0}^{N-1}\zeta_N^{jc}\left(\sum_{t=0}^{N/r-1}\zeta_N^{jtr}\right)|j\rangle = \frac{1}{\sqrt{r}}\sum_{k=0}^{r-1}\zeta_N^{ckN/r}|k\cdot N/r\rangle}$$

4) Measure, the value kN/r can be used to determine r. (Repeat if necessary).

Use of Periodicity Finding?

The quantum algorithm for periodicity finding works for a “black box function” F as long as it has the right properties (F is periodic, and unique within its period).

You can prove that any classical algorithm requires $\Theta(\text{poly}(r))$ time steps to solve the same problem.

We want to use this quantum subroutine to solve **natural problems** that are defined without reference to a black box function. That is: we want to look at explicit functions F.

Bad Example: The function $F(x) = x \text{ MOD } r$ has the right characteristics, but is easy classically.

A Hard Periodic Function

Take a (large) integer N, and an element $x \in \{0,1,\ldots,N-1\}$ with gcd(N,x)=1 (such that x has an inverse mod N).

The function $F: t \mapsto x^t \bmod N$ will be 'proper periodic'.

As F(0)=1, F(1)=x,…; F(r)=F(0)=1 shows that $x^r=1 \bmod N$.

With the quantum algorithm for period finding, we can efficiently solve the problem:
"Given N and x, determine r such that $x^r=1 \bmod N$".

Classically, this appears to be a hard problem.

Side Comments

- For the quantum algorithm to work, we have to efficiently implement the function $F: t \mapsto x^t \bmod N$.
- This can be done by the "repeated squaring trick":
  We can calculate $x \mapsto x^2 \mapsto x^4 \mapsto x^8 \bmod N$…. fast;
  hence we can calculate $x^t \bmod N$ in time poly(log t).

- Initially, we do not know the period r of $F:\mathbb{N}\to\{0,\ldots,N-1\}$, so we have to 'guess' how many F(0),F(1),F(2),… we want to evaluate in the superposition.
  You can show that $F(0),\ldots,F(\approx N)$ is sufficient.
  (Period finding is a robust algorithm: small mistakes in the function F do not matter.)

Factorizing by Period Finding

How to find a non-trivial factor of an integer N?

- Sketch of the algorithm using Period Finding mod N:
  1. Pick random x<N with gcd(x,N)=1
  2. Determine smallest r such that: $\boxed{x^r = 1 \bmod N}$
  3. If r is even (*), note that

     $$\boxed{\left(x^{r/2} - 1\right)\left(x^{r/2} + 1\right) = 0 \bmod N}$$

  4. Possible that $\boxed{x^{r/2} - 1}$ or $\boxed{x^{r/2} + 1}$ will share a non-trivial factor with N (use gcd for this) (*).
- (*) All this succeeds with high enough probability. Repeat if necessary.

Discrete Log Problem

- Let G be a finite group and take two elements Y and X, determine the power k such that $X^k=Y$, or "$\log_X(Y) = ?$"

- This takes place in the cyclic group $\langle X \rangle = \{1, X, X^2, \ldots\}$.

- Solving the Discrete Log Problem, also solves:
  - Diffie-Hellman problem
  - ElGamal Encryption (used for example in PGP)
  - Elliptic Curve Cryptography

Discrete Log Algorithm (1)

- First, determine order (M) of $\langle X \rangle = \{1, X, \ldots, X^{M-1}\}$.
- Next, create 'double superposition' and calculate

$$\boxed{\frac{1}{M}\sum_{s,t=0}^{M-1}\left|s,t,0\right\rangle \quad \mapsto \quad \frac{1}{M}\sum_{s,t=0}^{M-1}\left|s,t,Y^{s}\cdot X^{t}\right\rangle}$$

- "$X^k=Y$" tells us that this equals $\boxed{\frac{1}{M}\sum_{s,t}\left|s,t,X^{ks+t}\right\rangle}$
- Observe right register (assume outcome "X^c")

Discrete Log Algorithm (2)

- Measuring "c" gives $\boxed{\frac{1}{M}\sum_{s,t=0}^{M-1}\left|s,t,X^{ks+t}\right\rangle \mapsto \frac{1}{\sqrt{M}}\sum_{s=0}^{M-1}\left|s,c-ks,X^{c}\right\rangle}$
- Apply double QFT to two left registers

$$\boxed{\mapsto \frac{1}{M}\sum_{s=0}^{M-1}\sum_{i=0}^{M-1}\zeta_M^{is}\left|i\right\rangle \otimes \frac{1}{\sqrt{M}}\sum_{j=0}^{M-1}\zeta_M^{j(c-ks)}\left|j\right\rangle}$$

- This equals:

$$\boxed{= \quad \frac{1}{M\sqrt{M}}\sum_{i,j=0}^{M-1}\zeta_M^{jc}\left(\sum_{s=0}^{M-1}\zeta_M^{s(i-jk)}\right)\left|i,j\right\rangle \quad = \quad \frac{1}{\sqrt{M}}\sum_{j=0}^{M-1}\zeta_M^{jc}\left|jk,j\right\rangle}$$

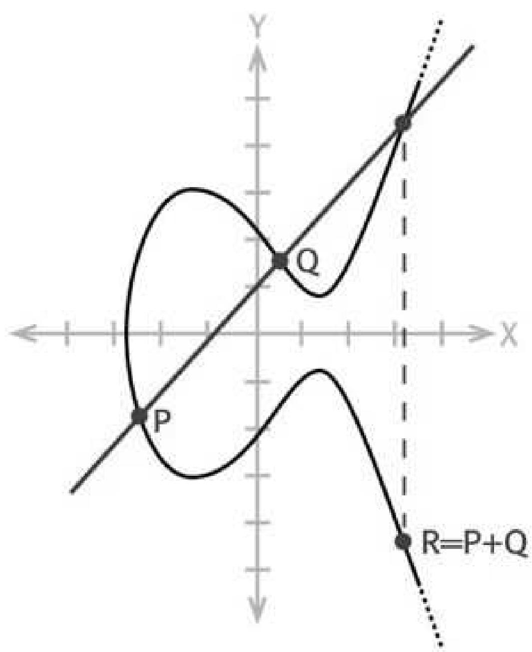
destructive interference for i≠jk mod M

Discrete Log Algorithm (3)

- Discrete Log Problem (X,Y) can be solved by:
  - Determine order X (let this be M)
  - Create superposition of $(s,t) \in \{0,1,\ldots,M-1\}^2$
  - Calculate function $s,t \rightarrow Y^s \cdot X^t$
  - Apply two Fouriers over $(s,t) \in \{0,1,\ldots,M-1\}^2$
  - Read out (s,t) register;
    the outcome will be (jk,j) for some random j
  - With high probability j is invertible mod M,
    if so, use (jk,j) to conclude k = jk/j mod M
  - This succeeds with high probability.

Elliptic Curve Cryptography

- Elliptic curve cryptography is based on the group that you can make of an elliptic curve (over a finite field).



The group operation + is defined in a nontrivial way, but it works.

The problem is: “Given P and Q, determine k such that k·P=Q.” Appears to be hard classically, but can be broken quantumly the same way logarithms are solved.

(Instead of multiplication mod M, we have addition over the curve.)