Breaking RSA encryption-Shor's Algorithm

PHYS/CSCI 3090

Prof. Alexandra Kolla

Alexandra.Kolla@Colorado.edu
ECES 122

Prof. Graeme Smith

Graeme.Smith@Colorado.edu
JILA S326

https://home.cs.colorado.edu/~alko5368/indexCSCI3090.html

Come see us!

- Alexandra Kolla/ Graeme Smith: Friday 3:00-4:00 pm, JILA X317.
- Ariel Shlosberg: Tu/Th 2:00-4:00pm, DUANG2B90 (physics help room)
- Steven Kordonowy: Th 11am-12pm, ECAE 124.
- Matteo Wilczak: Wednesday, 1-2pm, DUANG2B90 (physics help room)

Exam coming up!

- Midterm 2 March 18! (next week, on Wednesday)
- Focused on Chapter 6 and 3.

Last Class

- Finishing period Finding
- In class exercise without the offset

Today

- Finish Period finding in class
- Finish correctness argument
- Start on connections with Factoring

DFT

- Let $\omega = e^{2\pi i/N}$ a primitive N-th root of unity
- $f = \big(f(0), f(1), \dots, f(N-1)\big)$ a function
- The Discrete Fourier Transform of f is defined as

$$\mathrm{F} = \big(F(0), F(1), \dots, F(N-1)\big)$$

- Where $F(k) = \sum_n \omega^{kn} f(n)$

QFT

- Say we have n qubits. (So 2^{n} possible basis vectors).
- $\omega = \; e^{\frac{2\pi i}{2^n}}$
- $U_{FT}|x\rangle_n = \frac{1}{2^{\frac{n}{2}}}\sum_y \omega^{xy}|y\rangle_n$

QFT

- $f = \big(f(0), f(1), \dots, f(N-1)\big)$, $\mathrm{N} = 2^{\mathrm{n}}$
- $f = \sum_x f(x)|x\rangle_n$
- $U_{FT}(\sum_x f(x)|x\rangle_n) = \sum_y F(y)|y\rangle_n$
- $\boxed{F(y) = \frac{1}{2^{\frac{n}{2}}}\sum_x \omega^{xy} f(x)}$
- So $U_{FT}(\sum_x f(x)|x\rangle_n) = \frac{1}{2^{\frac{n}{2}}}\sum_y \sum_x \omega^{xy} f(x)\,|y\rangle_n$

The Algorithm

- Assume, for now, that r divides N.
- The superposition $\left(\sum_k |x_0 + kr\rangle\right)$ consists of N/r different multiples of r.
- So my state after collapse is $\frac{1}{\sqrt{\frac{N}{r}}}\left(\sum_{k=0\ to \frac{N}{r}} |x_0 + kr\rangle\right)$

The Algorithm

- Theorem:

Suppose the input to QFT is periodic with period r, for some r that divides N. Then the output will be a multiple of N/~~k~~r, and it is equally likely to be any of the r multiples of N/r.

- Now we repeat the experiment a few times and take GCD of all the indices returned, we get N/r (and thus r) w.h.p.

The Algorithm

- 4) Apply QFT to $\frac{1}{\sqrt{\frac{N}{r}}}\sum_j |x_0 + jr\rangle$. First assume $x_0 = 0$.

→ arbitrary number

assume $N/r \in \mathbb{N}$

- Claim (ex. in class):

If $|a\rangle = \frac{1}{\sqrt{\frac{N}{r}}}\left(\sum_j |jr\rangle\right)$ then

$$U_{FT}(|a\rangle) = |\beta\rangle = \boxed{\frac{1}{\sqrt{r}}\left(\sum_{j=0\ to\ r-1} \left|\frac{jN}{r}\right\rangle\right)}$$

Sums of roots of unity.

What is the sum $1+\omega^{jr}+\omega^{2jr}+\omega^{3jr}+\cdots+\omega^{(\frac{N}{r}-1)jr}$, where $\omega \quad = e^{2\pi i/N}$

$1+\omega^{jr}+\omega^{2jr}+\omega^{3jr}+\cdots+\omega^{\left(\frac{N}{r}-1\right)jr}=\frac{\mathrm{N}}{\mathrm{r}}$ if $jr=0 \; mod \; N$

$1+\omega^{jr}+\omega^{2jr}+\omega^{3jr}+\cdots+\omega^{\left(\frac{N}{r}-1\right)jr}=0$ otherwise

$$|a\rangle = \sum_{j=0}^{N/r-1} \sqrt{r/N}\, |jr\rangle = \sum_{j=0}^{N/r-1} \sqrt{\frac{r}{N}}\, |jr\rangle + \sum_{x} 0 \cdot |x\rangle$$

$$\stackrel{?}{=} \sum_{x} f(x)\, |x\rangle \qquad f_r(x) = \begin{cases} 0 & x \not\equiv 0 \bmod r \\ \sqrt{r/N} & \text{ow} \end{cases}$$

claim : $|a\rangle = \sum f_r(x)\,|x\rangle = \sum_{x=0 \bmod r} f_r(x)\,|x\rangle +$

$$\cancel{\boxed{\sum_{x \neq 0 \bmod r} f_r(x)\,|x\rangle}} = \sum_{j=0}^{N/r-1} \sqrt{r/N}\, |jr\rangle$$

$$U_{FT}\left(\sum f_r(x)\,|x\rangle\right) = \sum F_r(y)\,|y\rangle, \qquad F_r(y) = \sum_j \omega^{jy} f_r(j)$$

$$f_r(\ell) = \begin{cases} 0 & \text{ow} \\ \sqrt{\frac{r}{N}} & \text{if } \ell = 0 \bmod r \end{cases}$$

$$F_r(y) = \frac{1}{\sqrt{N}} \sum_{\ell} f_r(\ell)\, \omega^{\ell y}$$

$$F_r(y) = \frac{1}{\sqrt{N}} \cdot \sqrt{\frac{r}{N}} \sum_{i=0}^{N/r - 1} \omega^{iry} = \frac{\sqrt{r}}{N}\left(1 + \omega^{yr} + \omega^{2yr} + \dots\right)$$

$(\ell = ir)$

$$= \begin{cases} 0 & \text{if } yr \neq 0 \bmod N \\ \frac{\sqrt{r}}{N} \cdot \frac{N}{r} & \text{if } yr = 0 \bmod N \\ = 1/\sqrt{r} & \text{for } y = i \cdot N/r \end{cases}$$

$$\sum F_r(y)\,|y\rangle = \frac{1}{\sqrt{r}} \sum_{i=0}^{r-1} |i N/r\rangle$$

Some probability.

If I take a sample from the uniform distribution on {0,1...,r-1},
what is the probability (at most) that the number I get is a multiple of an integer j?

A) 0

B) 1/j

C) 1

D) I don't know

The Algorithm

- Lemma:

Suppose I take s independent samples drawn uniformly from $0, \frac{N}{r}, \ldots, \frac{(r-1)N}{r}$. Then with probability at least $1 - \frac{r}{2^s}$, the GCD of these samples is N/r.

e.g.: $N/r = 3$ $\quad 4 \cdot N/r \quad 8 \cdot N/r$ (bad)

$r = 100$

Some more probability.

Suppose I flip s coins, each of which has probability of heads *less* than p.
What is an upper bound on the the probability that all of them come out heads?

(the smallest)

A) p

B) p^s

C) $1 - p$

D) $1 - p^s$

Some more more probability.

Suppose I have a set of r bad events, $\{E_j\}_{j=0}^{r-1}$, and each of the E_j happens with probability at most p. What is and upper bound on the probability that at least one ~~none of~~ the bad events happen?

E_j might be dependent

A) p

B) rp

C) $1-p$

D) p^r

union bound

$\boxed{\Pr(\cup E_j) = ?} \leq \sum \Pr[E_j]$

The Algorithm

- Lemma:

Suppose I take s independent samples drawn uniformly from $0, \frac{N}{r}, \ldots, \frac{(r-1)N}{r}$. Then with probability at least $1 - \frac{r}{2^s}$, the GCD of these samples is N/r.