PS – due Tue

Next assignment – due week later.

$j = j_{n-1} 2^{n-1} + \cdots + j_0 2^0$

$\boxed{\text{Ph 452}}$ Lecture 16

Clarification: $\frac{jk}{2^n} = j_{n-1}(0.k_0) + \cdots + j_0(0.k_{n-1}\cdots k_0)$

Example: $n=3$, $j=2$, $k=3$:

$\frac{2\cdot 3}{2^3} = 0(0.1) + 1(0.10) + 1(0.000)$

$= 0.11 = 1/2 + 1/4 = 3/4 = 6/8$ ✓

101 Quick Review

Factoring N (n digits)

$O(n^3)$ • Prime-power testing

$O(n^2)$ • gcd

• order-finding

$O(n^2)$ • $\boxed{\text{Quantum Fourier Transform}}$ ← only quantum part (QFT)

? • Continued fraction Algorithm (CFA)

? • Oracle realization: modular exponentiation

Continued Fraction algorithm:

$$\frac{31}{13} = 2 + \frac{5}{13}$$

$$= 2 + \frac{1}{\frac{13}{5}}$$

$$= 2 + \frac{1}{2 + \frac{3}{5}}$$

$$= 2 + \frac{1}{2 + \frac{1}{\frac{5}{3}}}$$

$$= 2 + \frac{1}{2 + \frac{1}{1 + 2/3}}$$

$$= 2 + \frac{1}{2 + \frac{1}{1 + \frac{1}{3/2}}}$$

$$= 2 + \frac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{2}}}}$$

Generally, write

$$[a_0, \ldots, a_N] = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cdots + \cfrac{1}{a_N}}}$$

If $\varphi = \frac{k}{r}$, where k, r have n digits,

$O(n)$ split-and-invert steps,

each requires $O(n^2)$ basic arithmetic ops

Ph 45a Lecture 16

Modular exponentiation

Order-finding oracle:

$$|i\rangle|j\rangle \mapsto |i\rangle|j \oplus a^i \bmod N\rangle$$

Factoid: $(a \bmod N)(b \bmod N) \bmod N = ab \bmod N$

E.g. $(6 \bmod 4)(2 \bmod 4) \bmod 4 = 2 \cdot 2 \bmod 4 = 0$

$6 \cdot 2 \bmod 4 = 12 \bmod 4 = 0$

Let $i = i_{n-1} \cdots i_0 = i_{n-1} 2^{n-1} + i_{n-2} 2^{n-2} + \cdots + i_0 2^0$

$$a^i \bmod N = \left(a^{i_{n-1} 2^{n-1}} \bmod N\right)\left(a^{i_{n-2} 2^{n-2}} \bmod N\right) \cdots \left(a^{i_0 2^0} \bmod N\right) \bmod N$$

$O(n)$ multiplications $\times$ $O(n^2)$ ops/multiplication $\to O(n^3)$

But how hard to precompute $a^2 \bmod N$, $a^4 \bmod N$, $a^8 \bmod N$, ... $a^{2^{n-1}} \bmod N$?

Note: Each multiplication is $O(n^2)$, but their complexity adds to this.

$a^2 \bmod N$ $\quad O(n^2)$

$a^4 \bmod N = (a^2 \bmod N)(a^2 \bmod N)$ $\quad O(n^2)$

$a^8 \bmod N = (a^4 \bmod N)(a^4 \bmod N)$ $\quad O(n^2)$

$\vdots$

$O(n)$ multiplications $\times$ $O(n^2)$ ops/multiplication $\to O(n^3)$

$O(n^3)$ reversible gates: $|i\rangle|0\rangle \to |i\rangle|a^i \bmod N\rangle$!

Ph 452 Lecture 16

Factoring: parting thoughts

Shor; 1997: $O(n^2 \log n \log\log n)$ → more efficient classical steps

Kitaev: Different factoring algorithm based on "phase estimation" (PS)

Cleve & Watrous 2000: $O(\log n)$ depth circuit, $O(n^5 \log^2 n)$ size

Classical Number field Sieve (e.g., Pollard 1988) $e^{[n^{1/3}(\log^{2/3} n - O(1))]}$

Factoring: cracks RSA cryptosystem

Discrete-log: given a and b ($= a^s \bmod N$), find s

→ Cracks Diffie Hellman

Records: 5/21/2007: $2^{1039} - 1$ factored in crypto-world.com

"~100 Opteron 2.2 GHz + 4 GB RAM" years

$\boxed{\text{Ph 452}}$ Lecture 16

Quantum Algorithms not based on QFT or Grover's Alg (all are query algorithms)

① **Ordered Search**: N bits: 00000 1111 1111 1

Classical: $\log_2 N$ (optimal)

Q. lower bound: $\frac{1}{\pi}\ln N + O(1) \approx 0.221 \log_2 N$ [Hoyer, Neerbek, Shi 02]

Q. upper bound: $4\log_{6.05} N \approx 0.433 \log_2 N$ [Childs, Landahl, Parrillo 07]

Found by computer search

② **Element Distinctness**: $N \in \mathbb{N}$, $\Sigma = \mathbb{Z}_N$, $Y = \mathbb{B}$, $X = \Sigma^N$

$$g(X) = \begin{cases} 1 & x_i = x_j \text{ for some } i,j \\ 0 & \text{otherwise} \end{cases}$$

Classical: $\Theta(N)$

Q. lower bound: $\Omega(N^{2/3})$ [Shi, 02]

Q. upper bound: $O(N^{2/3})$ [Ambainis, 03]

Idea: Johnson graph:

vertices: subsets of $\mathbb{Z}_N$

edges: if vertices differ in 1 variable

Example: $N = 3$

$V_1 = 0$ $V_4 = 01$ $V_7 = 012$

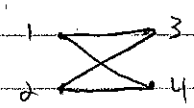
$V_2 = 1$ $V_5 = 02$ $V_8 = \emptyset$

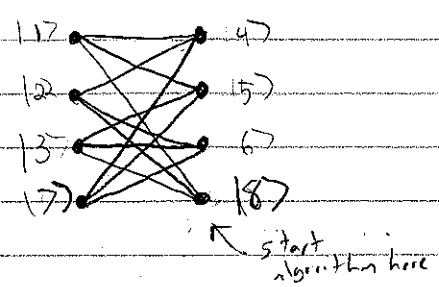
$V_3 = 2$ $V_6 = 12$

Example: $N = 2$

$V_1 = 0$ $V_3 = 01$

$V_2 = 1$ $V_4 = \emptyset$





"mark" vertices containing $x_i = x_j$ (already queried)

Quantum walk until marked vertex found

(Uses more structure than Grover Search)

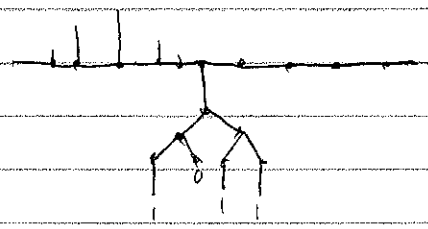
Ph 432 Lecture 16

3 NAND Trees

Classical : $\Theta(N^{0.753})$

Q. Lower bound : $\Omega(N^{1/2})$ [Barnum, Saks, Szegedy 2003]

Q. Upper bound : $O(N^{1/2+O(\frac{1}{\sqrt{\log N}})})$ [Ambainis 07]

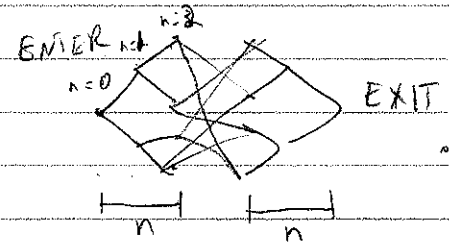


[Farhi, Goldstone, Gutmann 07]

Transmits/reflects whp if NAND Tree = 0/1.

4 Welded binary trees

[Childs, Cleve, Deotto, Farhi, Gutmann, Spielman 02]



- All non-root nodes have 3 neighbors
- $n+1$ bits suffice to label a vertex — $2n$ bits used by oracle
- $N = 2(2^{n+1}-1)$ vertices
- Oracle returns neighbors of a vertex → $2n$-bit strings used to label vertices

Classical : $\Theta(N)$

Q. lower bound : ?? (I don't know)

Q. upper bound : $O(\log N)$

Got this far