

# COL863: Quantum Computation and Information

Ragesh Jaiswal, CSE, IIT Delhi

## Quantum Computation: Order finding

# Quantum Computation

## Phase estimation $\rightarrow$ Order-finding

### Order finding

Given co-prime integers  $N > x > 0$ , compute the order of  $x$  modulo  $N$ .

- Consider the operator  $U$  that has the following behaviour:

$$U|y\rangle \equiv \begin{cases} |xy \pmod{N}\rangle & \text{if } 0 \leq y \leq N-1 \\ |y\rangle & \text{if } N \leq y \leq 2^L-1 \end{cases}$$

- Exercise summary: Let  $|u_s\rangle \equiv \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{-(2\pi i)\frac{sk}{r}} |x^k \pmod{N}\rangle$  be an eigenstate of  $U$ . Then  $U|u_s\rangle = e^{(2\pi i)\frac{s}{r}} |u_s\rangle$
- Main idea for determining  $r$ : We will use **phase estimation** to get an estimate on  $\frac{s}{r}$  and then obtain  $r$  from it.
  - How do we implement controlled  $U^{2^j}$ ? **Modular exponentiation**
  - How do we prepare an eigenstate  $|u_s\rangle$ ?
    - We work with  $|1\rangle$  as the first register since  $\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle = |1\rangle$ .
- So, we will argue that for each  $0 \leq s \leq r-1$ , we will obtain an estimate of  $\varphi \approx \frac{s}{r}$  accurate to  $2L+1$  bits with probability at least  $\frac{(1-\epsilon)}{r}$ .
  - Question: How do we extract  $r$  from this? **Continued fractions**

# Quantum Computation

## Digression: Continued fractions

### Continued fraction

A finite simple continued fraction is defined by a collection of positive integers  $a_0, \dots, a_N$ :

$$[a_0, \dots, a_N] \equiv a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_N}}}}$$

The  $n^{\text{th}}$  convergent ( $0 \leq n \leq N$ ) of this continued fraction is defined to be  $[a_0, \dots, a_n]$ .

- Theorem: Suppose  $x \geq 1$  is a rational number. Then  $x$  has a representation as a continued fraction,  $x = [a_0, \dots, a_N]$ . This may be found by the **continued fraction algorithm**.
- Exercise: Find the continued fraction expansion of  $\frac{31}{13}$ .
- Question: What is the running time for the continued fractions algorithm for any given rational number  $\frac{p}{q} \geq 1$ ?

# Quantum Computation

## Digression: Continued fractions

### Continued fraction

A finite simple continued fraction is defined by a collection of positive integers  $a_0, \dots, a_N$ :

$$[a_0, \dots, a_N] \equiv a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_N}}}}$$

The  $n^{\text{th}}$  convergent ( $0 \leq n \leq N$ ) of this continued fraction is defined to be  $[a_0, \dots, a_n]$ .

- Question: What is the running time for the continued fractions algorithm for any given rational number  $\frac{p}{q} \geq 1$ ?
- Theorem: Let  $a_0, \dots, a_N$  be a sequence of positive numbers. Then  $[a_0, \dots, a_n] = \frac{p_n}{q_n}$ , where  $p_n$  and  $q_n$  are real numbers defined inductively by  $p_0 \equiv 0$ ,  $q_0 \equiv 1$ ,  $p_1 \equiv 1 + a_0 a_1$ ,  $q_1 \equiv a_1$ , and for  $2 \leq n \leq N$ ,

$$p_n \equiv a_n p_{n-1} + p_{n-2}$$

$$q_n \equiv a_n q_{n-1} + q_{n-2}$$

In the case when  $a_j$  are positive integers, so too are  $p_j$  and  $q_j$  and moreover  $q_n p_{n-1} - p_n q_{n-1} = (-1)^n$  for  $n \geq 1$  which implies that  $\gcd(p_n, q_n) = 1$ .

# Quantum Computation

## Digression: Continued fractions

### Continued fraction

A finite simple continued fraction is defined by a collection of positive integers  $a_0, \dots, a_N$ :

$$[a_0, \dots, a_N] \equiv a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_N}}}}$$

The  $n^{\text{th}}$  convergent ( $0 \leq n \leq N$ ) of this continued fraction is defined to be  $[a_0, \dots, a_n]$ .

- **Question:** What is the running time for the continued fractions algorithm for any given rational number  $\frac{p}{q} \geq 1$ ?
  - Let  $[a_0, \dots, a_N] = \frac{p}{q} \geq 1$  with  $L = \lceil \log p \rceil$  and let  $p_n, q_n$  be as defined in the theorem.
  - **Observation:**  $p_n, q_n$  are increasing with  $p_n \geq 2p_{n-2}, q_n \geq 2q_{n-2}$ .
- **Theorem:** Let  $a_0, \dots, a_N$  be a sequence of positive numbers. Then  $[a_0, \dots, a_n] = \frac{p_n}{q_n}$ , where  $p_n$  and  $q_n$  are real numbers defined inductively by  $p_0 \equiv 0, q_0 \equiv 1, p_1 \equiv 1 + a_0 a_1, q_1 \equiv a_1$ , and for  $2 \leq n \leq N$ ,

$$p_n \equiv a_n p_{n-1} + p_{n-2}$$

$$q_n \equiv a_n q_{n-1} + q_{n-2}$$

In the case when  $a_j$  are positive integers, so too are  $p_j$  and  $q_j$  and moreover  $q_n p_{n-1} - p_n q_{n-1} = (-1)^n$  for  $n \geq 1$  which implies that  $\gcd(p_n, q_n) = 1$ .

# Quantum Computation

## Digression: Continued fractions

### Continued fraction

A finite simple continued fraction is defined by a collection of positive integers  $a_0, \dots, a_N$ :

$$[a_0, \dots, a_N] \equiv a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_N}}}}$$

The  $n^{\text{th}}$  convergent ( $0 \leq n \leq N$ ) of this continued fraction is defined to be  $[a_0, \dots, a_n]$ .

- Question: What is the running time for the continued fractions algorithm for any given rational number  $\frac{p}{q} \geq 1$ ?
  - Let  $[a_0, \dots, a_N] = \frac{p}{q} \geq 1$  with  $L = \lceil \log p \rceil$  and let  $p_n, q_n$  be as defined in the theorem.
  - Observation:  $p_n, q_n$  are increasing with  $p_n \geq 2p_{n-2}, q_n \geq 2q_{n-2}$ .
  - This implies that  $2^{\lfloor N/2 \rfloor} \leq q \leq p$ . So,  $N = O(L)$  and the running time of algorithm is  $O(L^3)$ .
- Theorem: Let  $a_0, \dots, a_N$  be a sequence of positive numbers. Then  $[a_0, \dots, a_n] = \frac{p_n}{q_n}$ , where  $p_n$  and  $q_n$  are real numbers defined inductively by  $p_0 \equiv 0, q_0 \equiv 1, p_1 \equiv 1 + a_0 a_1, q_1 \equiv a_1$ , and for  $2 \leq n \leq N$ ,  
$$p_n \equiv a_n p_{n-1} + p_{n-2}; \quad q_n \equiv a_n q_{n-1} + q_{n-2}$$

In the case when  $a_j$  are positive integers, so too are  $p_j$  and  $q_j$  and moreover  $q_n p_{n-1} - p_n q_{n-1} = (-1)^n$  for  $n \geq 1$  which implies that  $\gcd(p_n, q_n) = 1$ .

# Quantum Computation

## Digression: Continued fractions

### Continued fraction

A finite simple continued fraction is defined by a collection of positive integers  $a_0, \dots, a_N$ :

$$[a_0, \dots, a_N] \equiv a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_N}}}}$$

The  $n^{\text{th}}$  convergent ( $0 \leq n \leq N$ ) of this continued fraction is defined to be  $[a_0, \dots, a_n]$ .

- Theorem: Let  $x$  be a rational number and suppose  $\frac{p}{q}$  is a rational number such that  $|\frac{p}{q} - x| \leq \frac{1}{2q^2}$ . Then  $\frac{p}{q}$  is a convergent of the continued fraction for  $x$ .



# Quantum Computation

## Digression: Continued fractions

### Theorem

Let  $x$  be a rational number and suppose  $\frac{p}{q}$  is a rational number such that  $|\frac{p}{q} - x| \leq \frac{1}{2q^2}$ . Then  $\frac{p}{q}$  is a convergent of the continued fraction for  $x$ .

### Proof sketch

- Let  $\frac{p}{q} = [a_0, \dots, a_n]$  and let  $p_j, q_j$  as defined in the previous theorem so that  $\frac{p}{q} = \frac{p_n}{q_n}$ .
- Define  $\delta$  by the equation:

$$x \equiv \frac{p_n}{q_n} + \frac{\delta}{2q_n^2}, \text{ so that } |\delta| \leq 1.$$

- Define  $\lambda$  by

$$\lambda \equiv 2 \left( \frac{q_n p_{n-1} - p_n q_{n-1}}{\delta} \right) - \frac{q_{n-1}}{q_n}$$

# Quantum Computation

## Digression: Continued fractions

### Theorem

Let  $x$  be a rational number and suppose  $\frac{p}{q}$  is a rational number such that  $|\frac{p}{q} - x| \leq \frac{1}{2q^2}$ . Then  $\frac{p}{q}$  is a convergent of the continued fraction for  $x$ .

### Proof sketch

- Let  $\frac{p}{q} = [a_0, \dots, a_n]$  and let  $p_j, q_j$  as defined in the previous theorem so that  $\frac{p}{q} = \frac{p_n}{q_n}$ .
- Define  $\delta$  by the equation:  $x \equiv \frac{p_n}{q_n} + \frac{\delta}{2q_n^2}$ , so that  $|\delta| \leq 1$ .
- Define  $\lambda$  by  $\lambda \equiv 2 \left( \frac{q_n p_{n-1} - p_n q_{n-1}}{\delta} \right) - \frac{q_{n-1}}{q_n}$
- Claim 1:  $x = \frac{\lambda p_n + p_{n-1}}{\lambda q_n + q_{n-1}}$  and therefore  $x = [a_0, \dots, a_n, \lambda]$ .

# Quantum Computation

## Digression: Continued fractions

### Theorem

Let  $x$  be a rational number and suppose  $\frac{p}{q}$  is a rational number such that  $|\frac{p}{q} - x| \leq \frac{1}{2q^2}$ . Then  $\frac{p}{q}$  is a convergent of the continued fraction for  $x$ .

### Proof sketch

- Let  $\frac{p}{q} = [a_0, \dots, a_n]$  and let  $p_j, q_j$  as defined in the previous theorem so that  $\frac{p}{q} = \frac{p_n}{q_n}$ .
- Define  $\delta$  by the equation:  $x \equiv \frac{p_n}{q_n} + \frac{\delta}{2q_n^2}$ , so that  $|\delta| \leq 1$ .
- Define  $\lambda$  by  $\lambda \equiv 2 \left( \frac{q_n p_{n-1} - p_n q_{n-1}}{\delta} \right) - \frac{q_{n-1}}{q_n}$ .
- Claim 1:  $x = \frac{\lambda p_n + p_{n-1}}{\lambda q_n + q_{n-1}}$  and therefore  $x = [a_0, \dots, a_n, \lambda]$ .
- Claim 2:  $\lambda = \frac{2}{\delta} - \frac{q_{n-1}}{q_n} > 2 - 1 > 1$  which further implies that  $\lambda = [b_0, \dots, b_m]$  and  $x = [a_0, \dots, a_n, b_0, \dots, b_m]$ .
- This completes the proof of the theorem. □

# Quantum Computation

## Phase estimation $\rightarrow$ Order-finding

### Order finding

Given co-prime integers  $N > x > 0$ , compute the order of  $x$  modulo  $N$ .

- Consider the operator  $U$  that has the following behaviour:

$$U|y\rangle \equiv \begin{cases} |xy \pmod{N}\rangle & \text{if } 0 \leq y \leq N-1 \\ |y\rangle & \text{if } N \leq y \leq 2^L-1 \end{cases}$$

- Exercise summary: Let  $|u_s\rangle \equiv \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{-(2\pi i) \frac{sk}{r}} |x^k \pmod{N}\rangle$  be an eigenstate of  $U$ . Then  $U|u_s\rangle = e^{(2\pi i) \frac{s}{r}} |u_s\rangle$
- Main idea for determining  $r$ : We will use **phase estimation** to get an estimate on  $\frac{s}{r}$  and then obtain  $r$  from it.
  - How do we implement controlled  $U^{2^j}$ ? **Modular exponentiation**
  - How do we prepare an eigenstate  $|u_s\rangle$ ?
    - We work with  $|1\rangle$  as the first register since  $\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle = |1\rangle$ .
- So, we will argue that for each  $0 \leq s \leq r-1$ , we will obtain an estimate of  $\varphi \approx \frac{s}{r}$  accurate to  $2L+1$  bits with probability at least  $\frac{(1-\epsilon)}{r}$ .
  - Question: How do we extract  $r$  from this? **Continued fractions**
  - Question: Are we guaranteed to get  $r$  using continued fractions? What could go wrong?

# Quantum Computation

Phase estimation  $\rightarrow$  Order-finding

## Order finding

Given co-prime integers  $N > x > 0$ , compute the order of  $x$  modulo  $N$ .

- We obtain  $\varphi \approx \frac{s}{r}$  for some  $0 \leq s \leq r - 1$  and then we use continued fractions to obtain  $s', r'$  such that  $s'/r' = s/r$ .
- The problem is  $r'$  may not equal  $r$ . One such case is when  $s = 0$ . This, however, is a small probability event.
- Claim: Suppose we repeat twice and obtain  $s'_1, r'_1$  and  $s'_2, r'_2$ . If  $s_1$  and  $s_2$  are co-prime, then  $r = \text{lcm}(r'_1, r'_2)$ .

# Quantum Computation

Phase estimation  $\rightarrow$  Order-finding

## Order finding

Given co-prime integers  $N > x > 0$ , compute the order of  $x$  modulo  $N$ .

- We obtain  $\varphi \approx \frac{s}{r}$  for some  $0 \leq s \leq r - 1$  and then we use continued fractions to obtain  $s', r'$  such that  $s'/r' = s/r$ .
- The problem is  $r'$  may not equal  $r$ . One such case is when  $s = 0$ . This, however, is a small probability event.
- Claim: Suppose we repeat twice and obtain  $r'_1$  and  $r'_2$  corresponding to  $s_1, s_2$ . If  $s_1$  and  $s_2$  are co-prime, then  $r = \text{lcm}(r'_1, r'_2)$ .
- Claim:  $\Pr[s_1 \text{ and } s_2 \text{ are co-prime}] \geq 1/4$ .

# Quantum Computation

Phase estimation  $\rightarrow$  Order-finding

## Order finding

Given co-prime integers  $N > x > 0$ , compute the order of  $x$  modulo  $N$ .

## Quantum Order-finding

1.  $|0\rangle |1\rangle$  (Initial state)
2.  $\rightarrow \frac{1}{2^{t/2}} \sum_{j=0}^{2^t-1} |j\rangle |1\rangle$  (Create superposition)
3.  $\rightarrow \frac{1}{2^{t/2}} \sum_{j=0}^{2^t-1} |j\rangle |x^j \pmod{N}\rangle$  (Apply  $U_{x,N}$ )  
 $\approx \frac{1}{\sqrt{r}2^{t/2}} \sum_{s=0}^{r-1} \sum_{j=0}^{2^t-1} e^{(2\pi i)\frac{sj}{r}} |j\rangle |u_s\rangle$
4.  $\rightarrow \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |(s\tilde{r})\rangle |u_s\rangle$  (Apply inverse FT to 1<sup>st</sup> register)
5.  $\rightarrow (s\tilde{r})$  (Measure first register)
6.  $\rightarrow r$  (Use continued fractions algorithm)

- What is the size of the circuit that computes the order with high probability?

# Quantum Computation

Phase estimation  $\rightarrow$  Order-finding

## Order finding

Given co-prime integers  $N > x > 0$ , compute the order of  $x$  modulo  $N$ .

## Quantum Order-finding

1.  $|0\rangle |1\rangle$  (Initial state)
2.  $\rightarrow \frac{1}{2^{t/2}} \sum_{j=0}^{2^t-1} |j\rangle |1\rangle$  (Create superposition)
3.  $\rightarrow \frac{1}{2^{t/2}} \sum_{j=0}^{2^t-1} |j\rangle |x^j \pmod{N}\rangle$  (Apply  $U_{x,N}$ )  
 $\approx \frac{1}{\sqrt{r}2^{t/2}} \sum_{s=0}^{r-1} \sum_{j=0}^{2^t-1} e^{(2\pi i)\frac{sj}{r}} |j\rangle |u_s\rangle$
4.  $\rightarrow \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |(s\tilde{/}r)\rangle |u_s\rangle$  (Apply inverse FT to 1<sup>st</sup> register)
5.  $\rightarrow (s\tilde{/}r)$  (Measure first register)
6.  $\rightarrow r$  (Use continued fractions algorithm)

- What is the size of the circuit that computes the order with high probability?  $O(L^3)$



## Quantum Computation: Factoring

# Quantum Computation

Phase estimation → Order finding → Factoring

## Factoring

Given a positive composite integer  $N$ , output a non-trivial factor of  $N$ .

- We will solve the factoring problem by **reduction** to the order finding problem.
- Theorem 1: Suppose  $N$  is an  $L$  bit composite number, and  $x$  is a non-trivial solution to the equation  $x^2 = 1 \pmod{N}$  in the range  $1 \leq x \leq N$ , that is, neither  $x = 1 \pmod{N}$  nor  $x = -1 \pmod{N}$ . Then at least one of  $\gcd(x - 1, N)$  and  $\gcd(x + 1, N)$  is a non-trivial factor of  $N$  that can be computed using  $O(L^3)$  operations.
- Theorem 2: Suppose  $N = p_1^{\alpha_1} \dots p_m^{\alpha_m}$  is the prime factorisation of an odd composite positive integer. Let  $x$  be an integer chosen uniformly at random, subject to the requirement that  $1 \leq x \leq N - 1$  and  $x$  is co-prime to  $N$ . Let  $r$  be the order of  $x$  modulo  $N$ . Then

$$\Pr[r \text{ is even and } x^{r/2} \neq -1 \pmod{N}] \geq 1 - \frac{1}{2^m}.$$

# Quantum Computation

Phase estimation → Order finding → Factoring

## Factoring

Given a positive composite integer  $N$ , output a non-trivial factor of  $N$ .

## Quantum Factoring Algorithm

1. If  $N$  is even, return 2 as a factor.
2. Determine if  $N = a^b$  for integers  $a, b \geq 2$  and if so, return  $a$ .
3. Randomly choose  $1 \leq x \leq N - 1$ . If  $\gcd(x, N) > 1$ , then return  $\gcd(x, N)$ .
4. Use the Quantum order-finding algorithm to find the order  $r$  of  $x$  modulo  $N$ .
5. If  $r$  is even and  $x^{r/2} \not\equiv -1 \pmod{N}$ , then compute  $p = \gcd(x^{r/2} - 1, N)$  and  $q = \gcd(x^{r/2} + 1, N)$ . If either  $p$  or  $q$  is a non-trivial factor of  $N$ , then return that factor else return "Failure".

End