Quantum Algorithms
Lecture 25
Quantum algorithms for Abelian groups IV

Zhejiang University

Parallelized version of phase estimation. Applications.

Introduction

The phase estimation procedure (except for its last part — the continued fraction algorithm) can be realized by a quantum circuit of small depth.

This result is due to R. Cleve and J. Watrous, authors in this book created different proof.

Theorem

Eigenvalues of a unitary operator U can be determined with precision $\delta = 2^{-n}$ and error probability $\leq \varepsilon = 2^{-l}$ by an $O(n(l + logn))$-size, $O(logn + logl)$ -depth quantum circuit over the standard basis, with the additional gate $\Upsilon_m(U)$, $m = n + \log(l + logn) + O(1)$. This gate is used in the circuit only once.

Proof - ideas

Instead of applying the circuit $\Lambda(U^{p_t})[t, A] \cdots \Lambda(U^{p_1})[1, A]$, we compute $p = p(u_1, \ldots, u_t) = u_1 p_1 + \cdots + u_t p_t$, use p as the control parameter for the operator $\Upsilon_m(U)$, and uncompute p.

$$\Upsilon_m(U): |p\rangle \otimes |\xi\rangle \to |p\rangle \otimes U^p |\xi\rangle$$

Here $0 \leq p < 2^m$, and parameter $m = n + log(l + logn) + O(1)$ can be just considered as bound on our precision.

Proof - ideas

Authors optimize the sum calculation $p = \sum_{r=1}^{t} u_r p_r$, so we can use a circuit of size $O(ns) = O(n(l + logn))$ and depth $O(logn + logs) = O(logn + logl)$ for the addition of $2s$ n-digit numbers.

$t = 2ns$, so complexity of remaining parts on the algorithm is not exceeding the complexity of sum calculation.

Theorem - remark

Theorem does not imply that the algorithms for period finding and factoring can be fully parallelized. However, one can derive the following corollary.

We have not parallelized phase estimations and pre/post processing.

Corollary

Period finding and factoring can be performed by a uniform sequence of $O(n^3)$-size, $O((logn)^2)$-depth quantum circuits, with some classical pre-processing and post-processing. The pre-processing and post-processing are realized by uniform sequences of $O(n^3)$-size Boolean circuits.

Pre- and post-processing

Here we do classical pre-processing so that quantum circuit will already receive necessary values (definition allows to precompute the circuit for specific input and launch it as if no input is taken into consideration. Computation of the numbers $a^{2^j} \bmod q$ ($j = 0, \ldots, 2n-1$) is done in advance.

Classical post-processing cannot be omitted: continued fraction algorithm + Euclid's algorithm.

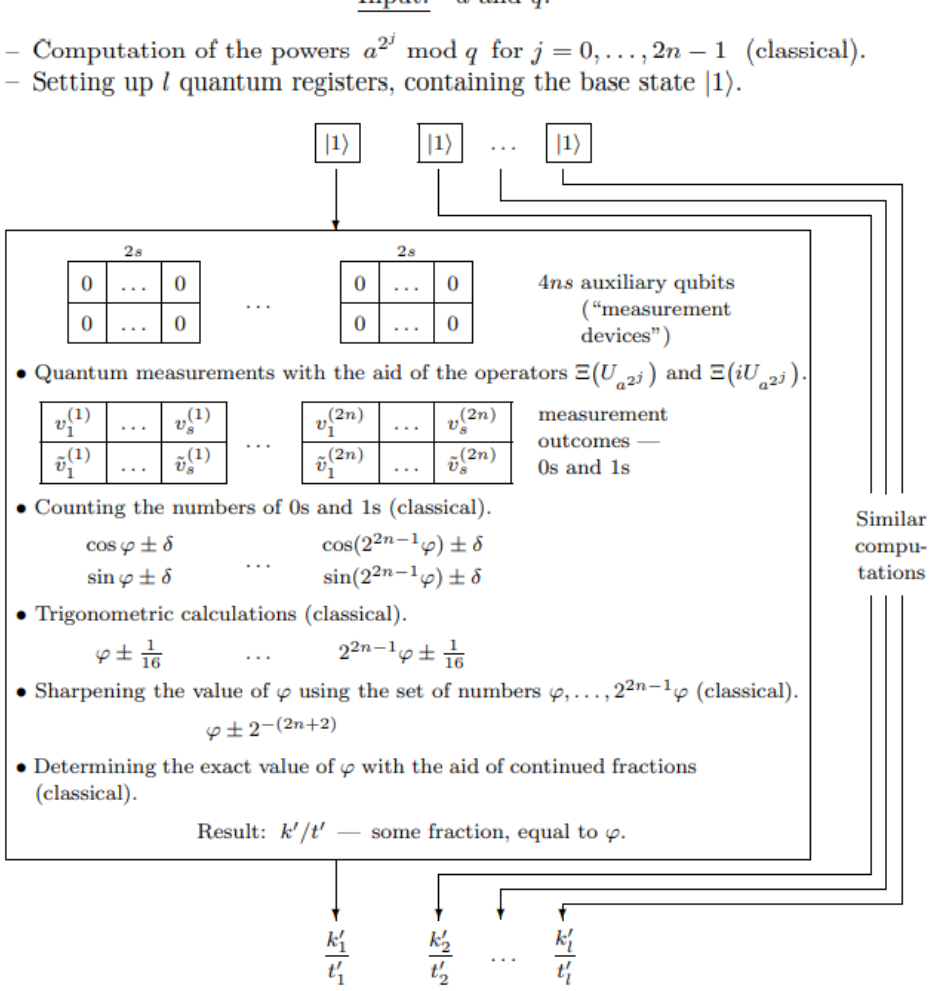

Table 13.1. General scheme of the period finding algorithm. Shown in a box is the phase estimation part.

Proof of corollary

If F and F^{-1} can be computed by Boolean circuits of size $\leq L$ and depth $\leq d$, then F can be realized by a reversible (quantum) circuit of size $O(L+n)$ and depth $O(d)$ using ancillas.

We realize $\Upsilon(U_a)$: $|p,x\rangle \rightarrow |p,(a^p x \; mod \; q)\rangle$

With pre-computed values of $(a^{2^j} \; mod q)$ and $(a^{-2^j} \; mod q)$, the computation of $(a^p x \; mod \; q)$ or $(a^{-p} x \; mod \; q)$ amounts to multiplying $O(n)$ numbers and calculating the residue $mod \; q$, which is done by a circuit of size $O(n^3)$ and depth $O((logn)^2)$.

Complexity remark

R. Cleve and J. Watrous also noticed that the depth can be decreased at the cost of increase in size. Indeed, the multiplication of $O(n)$ n-digit numbers can be performed with depth $O(logn)$ and size $O(n^5\ (\log n)^2)$; therefore the same bound applies to period finding and factoring.

It is fascinating that such classically complex tasks in quantum case have the main complexity component coming just from basic multiplication operation.

Realization of a unitary

Any unitary operator U on a fixed number of qubits can be realized with precision δ by a $poly(\log(1/\delta))$ -size, $poly(loglog(1/\delta))$ -depth circuit over the standard basis, using ancillas. There is a polynomial algorithm that constructs this circuit on the description of U.

Authors will try to simulate a circuit providing more general solution. Also with more precise complexity measures.

Lemma

The operator $\Upsilon_n\ (e^{2\pi i/2^n}) : |l\rangle \rightarrow e^{2\pi il/2^n}|l\rangle$ $(0 \leq l < 2^n)$ can be realized with precision $\delta = 2^{-n}$ by an $O(n^2 logn)$-size $O((logn)^2)$-depth circuit C_n over the standard basis, using ancillas. The circuit C_n can be constructed algorithmically in time $poly(n)$.

Lemma – step 1

Create the vector $|\eta\rangle = \sigma^z[1]H[1]|0^n\rangle$.

$$|\eta\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|2^{n-1}\rangle = \frac{1}{\sqrt{2^{n-1}}}\sum_{s=1}^{2^{n-1}}|\psi_{n,2s-1}\rangle$$

We get a superposition of all odd values of k.

$$|\psi_{n,k}\rangle = \frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}\exp\left(-2\pi i\frac{kj}{2^n}\right)|j\rangle$$

Lemma – step 2

Measure k with error probability $\leq \varepsilon = \delta^2/4$. To find k, it suffices to determine the phase $\phi_k = k/2^n$ with precision $\delta = 2^{-n}$. Such phase estimation is realized by a circuit of size $O(n^2)$ and depth $O(logn)$. The measured value should be odd, $k = 2s - 1$. (If it has happened to be even, set $k = 1$.)

$$|\psi_{n,k}\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} \exp\left(-2\pi i \frac{kj}{2^n}\right) |j\rangle$$

are eigenvectors of $X : |j\rangle \rightarrow |(j\ +\ 1)\ mod\ 2^n\rangle$,
$X|\psi_{n,k}\rangle = e^{2\pi i\phi_k}|\psi_{n,k}\rangle$, $\phi_k = k/2^n$

Lemma – step 3

Find $p = p(s, l)$ satisfying the equation $(2s - 1)p \equiv l(mod 2^n)$. The solution can be obtained as follows:

$$p \equiv -l \sum_{j=0}^{m-1} (2s)^j \equiv -l \prod_{r=1}^{t-1} \left(1 + (2s)^{2^r}\right) \mod 2^n, \quad m = 2^t, \quad t = \lceil \log_2 n \rceil$$

This calculation is done by a circuit of size $O(n^2 logn)$ and depth $O((logn)^2)$.

Lemma – step 4

Apply X^p to the n-bit register (which presumably contains $|\psi_{n,2s-1}\rangle$. This will effect the desired phase shift.

$$X^p|\psi_{n,2s-1}\rangle = e^{2\pi i((2s-1)p/2^n)}|\psi_{n,2s-1}\rangle$$

Because $(2s-1)p \equiv l(mod 2^n)$, we get

$$e^{2\pi il/2^n}|\psi_{n,2s-1}\rangle$$

Lemma – step 5

Reverse the computation done at Steps 1–3.

In such way we managed to multiply the amplitude of according state by $e^{2\pi i l/2^n}$ and return the system to initial states.

Theorem

Any circuit C of size L and depth d over a fixed finite basis C can be simulated with precision δ by an $O(Ln + n^2 logn)$-size $O(dlogn + (logn)^2)$-depth circuit $\tilde{C}$ over the standard basis (using ancillas), where $n = O(\log(L/\delta))$.

Proof of Theorem 8.3

Each gate of the original basis C can be replaced by a constant size circuit over the basis $Q \cup \{\Lambda(e^{i\phi})\colon \phi \in R\}$ (Q is standard basis, for proof see Chapter 8 for exact realization). The circuit C is transformed into a circuit C' of size $L' = O(L)$ and depth $d' = O(d)$ over the new basis. Each gate $\Lambda(e^{i\phi})$ can be approximated with precision $\delta' = \delta/(3L')$ by a gate of the form $\Lambda(e^{2\pi il/2^n})$, where $n = \log_2(1/\delta')$, and l is an integer. $\Lambda(e^{2\pi il/2^n})$ – special case of previous Lemma.

The resulting circuit suffices for the proof of Theorem 8.3 (which corresponds to the case $L = d = 1$).

Proof of Theorem for circuits

Most of the resource usage in Lemma can be attributed to solving the equation $kp \equiv l(mod 2^n)$. But this step is redundant if $k = 1$. In fact, the operator $\Upsilon_n\,(e^{2\pi i/2^n}) : |l\rangle \rightarrow e^{2\pi il/2^n}|l\rangle$ can be realized by applying

$$\Upsilon_n(X): |p, j\rangle \rightarrow |p, (j + p)\, mod 2^n\rangle$$

to the target state $|\psi_{n,1}\rangle$; this is done by a circuit of size $O(n)$ and depth $O(logn)$. Thus we need to create L' copies of the state $|\psi_{n,1}\rangle$ and use one copy per gate in the simulation of the circuit C'.

Theorem - sequence

Step 1. Create the state $|\psi_{n,0}\rangle = H^{\otimes n}|0^n\rangle$.

Step 2. Turn it into $|\psi_{n,1}\rangle = \Upsilon_n\,(e^{2\pi i/2^n})|\psi_{n,0}\rangle$ by the procedure of Lemma 13.4. This is done with precision $\delta' = 2^{-n} \leq \delta/3$. The corresponding circuit has size $O(n^2 logn)$ and depth $O((logn)^2)$.

Theorem - sequence

Step 3. Make L' copies of the state $|\psi_{n,1}\rangle$ out of one copy.

Step 4. Simulate the circuit C' with precision $\delta/3$, using one copy of $|\psi_{n,1}\rangle$ per gate.

Step 5. Reverse Steps 1–3.

For Step 3:

$$|\psi_{n,k}\rangle^{\otimes m} = W^{-1}\left(|\psi_{n,0}\rangle^{\otimes(m-1)} \otimes |\psi_{n,k}\rangle\right),$$

$$W : |x_1, \ldots, x_{m-1}, x_m\rangle \mapsto |x_1, \ldots, x_{m-1}, x_1 + \cdots + x_m\rangle$$

The operators W and W^{-1} are realized as addition of m n-digit numbers, which is done by a Boolean circuit of size $O(nm)$ and depth $O(logn + logm)$. In our case $m = L' = O(L)$.

We get in total $O(Ln + n^2 logn)$-size

$O(dlogn + (logn)^2)$-depth circuit

The hidden subgroup problem for $\mathbb{Z}^k$

Abelian group

A set A, together with an operation $\cdot$ that combines any two elements a and b of A to form another element of A, denoted $a \cdot b$. Abelian group axioms:

- For all a, b in A, the result of the operation $a \cdot b$ is also in A.
- For all a, b, and c in A, the equation $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ holds.
- There exists an element e in A, such that for all elements a in A, the equation $e \cdot a = a \cdot e = a$ holds.
- For each a in A there exists an element b in A such that $a \cdot b = b \cdot a = e$, where e is the identity element.
- For all a, b in A, $a \cdot b = b \cdot a$ (commutativity).

Problem for $\mathbb{Z}^k$

The algorithms discovered by Simon and Shor can be generalized to a rather broad class of problems connected with Abelian groups. The most general of these is the hidden subgroup problem for $\mathbb{Z}^k$, to which the hidden subgroup problem in an arbitrary finitely generated Abelian group G can be reduced.

For group $\mathbb{Z}^k$ each group member is represented as k integers.

Problem for $\mathbb{Z}^k$

A “hidden subgroup” $D \subseteq \mathbb{Z}^k$ has finite index: the order of the group $E = \mathbb{Z}^k / D$ does not exceed 2^n. Therefore $D \cong \mathbb{Z}^k$. From the computational viewpoint, D is given by a basis $(g1, \ldots, gk)$ whose binary representation has length $poly(k, n)$. Any such basis gives a solution to the problem. (The equivalence of two bases can be verified by a polynomial algorithm.)

A basis, also called an integral basis, is a subset such that every element of the group can be uniquely expressed as a linear combination of basis elements with integer coefficients. For instance, the integers with addition form a free abelian group with basis {1}.

Reduction from period finding

The problem of computing the period is a special case of the hidden subgroup problem in $\mathbb{Z}$. $D = \{m\, per_q\,(a)\colon m \in \mathbb{Z}\}$. This function is polynomially computable, hence an arbitrary polynomial algorithm for finding a hidden subgroup can be transformed into a polynomial algorithm for calculating the period.

The function $f\colon x \to a^x\ (mod q)$

$$f(x) = f(y) \iff x - y \in D$$

Reduction from discrete logarithm

The well-known problem of calculating the discrete logarithm can be reduced to the hidden subgroup problem for $\mathbb{Z}^2$. The smallest positive integer s such that $\zeta^s = a$, where ζ is a generator of the group $(Z/qZ)^*$, is called the discrete logarithm of a number a at base ζ.

$$f\colon (x1, x2) \to \zeta^{x1} a^{x2} mod q$$

$$D = \{(x1, x2) \in \mathbb{Z}^2\colon \zeta^{x1} a^{x2} \equiv 1 mod q\}$$

$$f(x) = f(y) \iff x - y \in D$$

HSP for G=$\mathbb{Z}^k$

Algorithm is analogous to the algorithm for the case $G = (\mathbb{Z}_2)^k$, but instead of the operator $H^{\otimes k}$ we use the procedure for measuring the eigenvalues. Instead of a basis for the group D we will look for a system of generators of the character group $E^* = Hom(E, U(1))$ (the transition from E^* to D is realized by a polynomial algorithm). The character

$$(g_1, \ldots, g_k) \mapsto \exp\left(2\pi i \sum_j \varphi_j g_j\right)$$

is determined by the set $\phi_1, \ldots, \phi_k$ of numbers modulo 1. These are rational numbers with denominators not exceeding $|E^*| \leq 2^n$.

HSP for G=$\mathbb{Z}^k$

Authors generalize the ideas of algorithms for Simon's problem and for period finding. Notice the different eigenvectors and eigenvalues compared to Period finding and new parameters to evaluate error probability to tune the parameters of the algorithm.

QFT

Quantum Fourier Transform also is mentioned. Matrix form:

$$\frac{1}{\sqrt{N}}\begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \cdots & \omega^{2N-2} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \cdots & \omega^{3N-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \omega^{N-1} & \omega^{2N-2} & \omega^{3N-3} & \cdots & \omega^{(N-1)(N-1)} \end{pmatrix}$$

QFT can be efficiently used for Period finding. Here $\omega = e^{2\pi i/N}$ is root of unity. The best quantum Fourier transform algorithms known (as of late 2000) require only $O(\mathrm{n}\log n)$ gates to achieve an efficient approximation.

HSP for G=$\mathbb{Z}^k$ - parameters

- $l = n + 3$ uniformly distributed random characters;
- precision δ - for each character element;
- the probability of error $\leq \varepsilon$
- $M = 2^m$ (integers between 0 to $M - 1$)
- n - number of qubits in second register
- Character values are obtained with precision $\delta = 2\beta$ and error probability $\leq \varepsilon = \frac{2}{M\beta}$
- $\delta \leq \frac{1}{2^{2n+1}}$
- $\varepsilon \leq \frac{1}{5kl}$

Pick M and β: $\delta = 2\beta \leq \frac{1}{2^{2n+1}}$, so $\beta \leq \frac{1}{2^{2n}}$; $\varepsilon = \frac{2}{M\beta} \leq \frac{1}{5kl}$,

so $M \geq \frac{10kl}{\beta}$; $M \geq 10k(n+3)2^{2n}$.

Complexity of the algorithm

We need $O(n)$ queries to the oracle, each query being of length $O(k(n + logk))$. The size of the quantum circuit is estimated as $O(kn^3)poly(logk, logn)$.

Thank you for your attention!