<table>
<thead>
<tr>
<th>項目</th>
<th>内容</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>An Introduction to Quantum Complexity Theory</td>
</tr>
<tr>
<td>Author(s)</td>
<td>Nishino, Tetsuro</td>
</tr>
<tr>
<td>Citation</td>
<td>物性研究 (1999), 72(3): 372-376</td>
</tr>
<tr>
<td>Issue Date</td>
<td>1999-06-20</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://hdl.handle.net/2433/96622">http://hdl.handle.net/2433/96622</a></td>
</tr>
<tr>
<td>Right</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>Departmental Bulletin Paper</td>
</tr>
<tr>
<td>Textversion</td>
<td>publisher</td>
</tr>
</tbody>
</table>

Kyoto University
An Introduction to Quantum Complexity Theory

Tetsuro Nishino

Department of Communications and Systems Engineering
The University of Electro - Communications
Chofu, Tokyo, 182-8585, Japan
e-mail: nishino@sw.cas.uec.ac.jp

Abstract In 1985, D. Deutsch introduced quantum Turing machines (QTMs for short) as Turing machines which can perform so called quantum parallel computations. Then, in 1994, P. Shor showed that QTM can factor integers with arbitrary small error probability in polynomial time. The quantum complexity theory is the computational complexity theory based on QTMs. In this paper, we first review the definitions of the QTM and major quantum complexity classes EQP, BQP, ZQP, etc. Then, we present the known results and major open questions in this field.

1 Introduction

Current computers are implemented based on Turing machine introduced by Alan Turing in 1937. Since Turing machine is a very simple and stable model of computation, it is used as a standard model in recursive function theory and computational complexity theory. Many results in complexity theory, however, suggests that deterministic Turing machines cannot efficiently solve hard combinatorial problems, such as NP-complete problems.

In 1985, David Deutsch introduced quantum Turing machines (QTMs for short) as Turing machines which can perform so called quantum parallel computations [2]. Then, in 1994, Peter Shor showed that QTM can factor integers with arbitrary small error probability in polynomial time [5]. Since it is widely believed that any deterministic Turing machines cannot factor integers in polynomial time, it is very likely that QTM is an essentially new model of computation. Many computer scientists are working hard in this area these days, and have obtained a lot of important results [1, 3, 5, 6, 7].

One tape cell of a Turing machine can contain a symbol 0 or 1, i.e. one bit of information. On the other hand, one tape cell of a QTM can be in an arbitrary superposition...
of the states 0 and 1, which is called one qubit (quantum bit) of information. Here, a superposition of the states 0 and 1 is represented by $\alpha|0\rangle + \beta|1\rangle$, where $|0\rangle$ and $|1\rangle$ are state vectors in some Hilbert space denoting the states corresponding to 0 and 1, respectively. $\alpha$ and $\beta$ are complex numbers such that $|\alpha|^2 + |\beta|^2 = 1$, and $\alpha$ ($\beta$) is called an amplitude of the state $|0\rangle$ ($|1\rangle$).

A computation of a QTM is a sequence of applications of unitary transformations to some qubits on its tape. After the computation, if we observe a tape cell in a superposition $\alpha|0\rangle + \beta|1\rangle$, we will see 0 (1) with probability $|\alpha|^2$ ($|\beta|^2$). Thus, if we observe a tape cell in a superposition $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, we will see 0 or 1 with equal probability 1/2. Namely, this tape cell is an ideal random bit. But, when we observe this tape cell, the superposition is completely destroyed.

The quantum complexity theory is the computational complexity theory based on QTMs. In this paper, we first review the definitions of the QTM and major quantum complexity classes. Then, we present the known results and major open questions in this field. For details, see [4] for example.

## 2 Quantum Turing Machines

We first review the definition of QTM which was formulated in [1]. Like an ordinary Turing machine, a quantum Turing machine consists of a finite control, an infinite tape, and a tape head.

**Definition 2.1** [1] A quantum Turing machine (QTM for short) is a 7-tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$, where $Q$ is a finite set of states, $\Gamma$ is a tape alphabet, $B \in \Gamma$ is a blank symbol, $\Sigma \subseteq \Gamma$ is an input alphabet, $\delta$ is a state transition function and is a mapping from $Q \times \Gamma \times \Gamma \times Q \times \{L, R\}$ to $C$ (the set of complex numbers), $q_0 \in Q$ is an initial state, and $F \subseteq Q$ is a set of final states.

An expression $\delta(p, a, b, q, d) = c$ represents the following: if $M$ in a state $p$ reads a symbol $a$ (let $C_1$ be this configuration of $M$), $M$ writes a symbol $b$ on the square under the tape head, changes the state into $q$, and moves the head on the square in the direction denoted by $d \in \{L, R\}$ (let $C_2$ be this configuration of $M$). Then, the complex number $c$ is called an amplitude of this event, and the probability that $M$ changes its configuration from $C_1$ to $C_2$ is defined to be $|c|^2$.

This state transition function $\delta$ defines a linear mapping in a linear space of superpositions of $M$'s configurations. This linear mapping is specified by the following matrix $M_\delta$. Each row and column of $M_\delta$ corresponds to a configuration of $M$. Let $C_1$ and $C_2$ be two configurations of $M$, then the entry corresponding to $C_2$ row and $C_1$ column of $M_\delta$ is $\delta$ evaluated at the tuple which transforms $C_1$ into $C_2$ in a single step. If no such tuple exists, the corresponding entry is 0. We call this matrix $M_\delta$ a time evolution matrix of $M$.

**Restriction:** For any QTM $M$, the time evolution matrix $M_\delta$ must be a unitary matrix.

Namely, if $M_\delta^\dagger$ is the transpose conjugate of $M_\delta$ and $I$ is the identity matrix, then the relation $M_\delta^\dagger M_\delta = M_\delta M_\delta^\dagger = I$ must be satisfied by $M_\delta$. 

---

"第6回「非平衡系の統計物理」シンポジウム"
Computation of \( M \) is an evolution process of a physical system defined by the unitary matrix \( M_6 \). Let \( |\psi(0)\rangle \) be an initial state of \( M \). If we denote the state of \( M \) at time \( s \) by \( |\psi(s)\rangle \), we have \( |\psi(\tau t)\rangle = M_6^\dagger |\psi(0)\rangle \) where \( \tau \) is the time required by \( M \) to execute a single step.

In quantum mechanics, observations from outside will change the state of the physical system. Thus, we can not observe the tape contents of a QTM from outside before the computation is terminated. When the computation is terminated, the tape contents will be observed as follows: if a QTM \( M \) in superposition \( \psi = \sum_i \alpha_i C_i \) is observed, configuration \( C_i \) is seen with probability \( |\alpha_i|^2 \), and the superposition of \( M \) is updated to \( C_i \). Especially, when a tape cell of a QTM is in a superposition \( \alpha|0\rangle + \beta|1\rangle \), we will see 0 (1) with probability \( |\alpha|^2 (|\beta|^2) \).

We may also perform a partial observation. Let us consider the case of a partial observation only on the first cell of the tape. Suppose the superposition was \( \psi = \sum_i \alpha_i C_i^0 + \sum_i \beta_i C_i^1 \), where \( C_i^0 \) (\( C_i^1 \)) are those configurations that have a 0 (1) in the first cell. In this case, if we observe the first cell, we will see 0 (1) with probability \( \frac{1}{|\sum_i \alpha_i|^2} (\frac{1}{|\sum_i \beta_i|^2}) \).

Moreover, if a 0 is observed, the new superposition is given by

\[
\frac{1}{|\sum_i \alpha_i|^2} \sum_i \alpha_i C_i^0,
\]

and if a 1 is observed, the new superposition is given by

\[
\frac{1}{|\sum_i \beta_i|^2} \sum_i \beta_i C_i^1.
\]

### 3 Quantum Complexity Classes

In computational complexity theory, a problem is considered to be a membership problem for a certain language. Let \( \Sigma \) be an alphabet. A membership problem for a language \( L \subseteq \Sigma^* \) is as follows: given a string \( x \in \Sigma^* \) decide whether \( x \in L \). A language \( L \) is in the class \( \text{P} \) (Polynomial time) if there exists a deterministic Turing machine with a distinguished acceptance tape cell and a polynomial \( p \) such that, given an arbitrary input string \( x \), whether \( x \in L \) is decided with probability 1 by observing the acceptance tape cell at time \( p(|x|) \). A quantum analogue of the class \( \text{P} \) is defined as follows.

**Definition 3.1**[1] A language \( L \) is in the class \( \text{EQP} \) (Exact Quantum Polynomial time) if there exists a QTM with a distinguished acceptance tape cell and a polynomial \( p \) such that, given an arbitrary input string \( x \), whether \( x \in L \) is decided with probability 1 by observing the acceptance tape cell at time \( p(|x|) \).

A language \( L \) is in the class \( \text{BPP} \) (Bounded error Probabilistic Polynomial time) if there exists a probabilistic Turing machine with a distinguished acceptance tape cell and a polynomial \( p \) such that, given an arbitrary input string \( x \), whether \( x \in L \) is decided with probability at least 2/3 by observing the acceptance tape cell at time \( p(|x|) \). A quantum analogue of the class \( \text{BPP} \) is defined as follows.

**Definition 3.2**[1] A language \( L \) is in the class \( \text{BQP} \) (Bounded error Quantum Polynomial time) if there exists a QTM with a distinguished acceptance tape cell and a
polynomial $p$ such that, given an arbitrary input string $x$, whether $x \in L$ is decided with probability at least $2/3$ by observing the acceptance tape cell at time $p(|x|)$.

A language $L$ is in the class $\text{ZPP}$ (Zero error Probabilistic Polynomial time) if there exists a probabilistic Turing machine with a distinguished acceptance tape cell and a halting cell, and a polynomial $p$ such that, given an arbitrary input string $x$, (1) if the halting cell is observed at time $p(|x|)$, 1 (which means the machine has been terminated) is seen with probability at least $1/2$, and (2) when 1 is observed at the halting cell, whether $x \in L$ is decided with probability 1 by observing the acceptance tape cell. A quantum analogue of the class ZPP is defined as follows.

**Definition 3.3** A language $L$ is in the class $\text{ZQP}$ (Zero error Quantum Polynomial time) if there exists a QTM with a distinguished acceptance tape cell and a halting cell, and a polynomial $p$ such that, given an arbitrary input string $x$, (1) if the halting cell is observed at time $p(|x|)$, 1 (which means the machine has been terminated) is seen with probability at least $1/2$, and (2) when 1 is observed at the halting cell, whether $x \in L$ is decided with probability 1 by observing the acceptance tape cell.

### 4 Open Problems

On the complexity classes introduced in the previous section, the following relations are known.

1. $\text{P} \subseteq \text{ZPP} \subseteq \text{BPP}$
2. $\text{QP} \subseteq \text{ZQP} \subseteq \text{BQP}$
3. $\text{P} \subseteq \text{QP}$
4. $\text{ZPP} \subseteq \text{ZQP}$
5. $\text{BPP} \subseteq \text{BQP}$

Major open questions concerning these relations are listed below.

1. $\text{BPP} \subseteq \text{QP}$ ?
2. $\text{NP} \subseteq \text{BQP}$ ?
3. $\text{BPP} = \text{BQP}$ ?

Here, NP is the class of languages accepted by nondeterministic Turing machines in polynomial time.

Let $R \subseteq \Sigma^* \times \Sigma^*$ be a relation. A function problem corresponding to $R$ is the following computation problem: given $x \in \Sigma^*$, find a string $y \in \Sigma^*$ such that $R(x, y)$ if such a string exists, and if such a string does not exist, return NO. For example, the factoring problem may be corresponded to the relation $R(x, y)$ which is defined as follows: for integers $x$ and $y$, $R$ holds if $y = p\#q$ for some integers $p$ and $q$ ($\#$ is a delimiter), and $x = p \times q$.

P. Shor showed that QTM can factor an integer with an arbitrary small error probability in polynomial time [5]. It is conjectured that any probabilistic Turing machine cannot perform the same thing.
References


