Quantum Turing Machines

Margherita Zorzi, PhD

Dipartimento di Informatica
Università di Verona

QC A.A. 2008/2009
Outline

1. Introduction
2. The Quantum Turing Machine: Formal Description
3. The Computational Power of QTM
4. Equivalence Results on Quantum Computational Models
5. Bibliography
The Quantum Turing Machine was defined by David Deutsch (1985) as precise model of a quantum physical computer.

There are two ways of thinking about QTM:
- the quantum physical analogue of a Probabilistic Turing Machine;
- computation as transformation in a space of complex superposition of configuration.

The QTM is the most general model for a computing device based on quantum physics.
Basic Definitions

Computable Numbers

Definition

1. A real number $x \in \mathbb{R}$ is *computable* iff there is a Deterministic Turing Machine which on input $1^n$ computes a binary representation of an integer $m \in \mathbb{Z}$ such that $|m/2^n - x| \leq 1/2^n$. Let $\tilde{\mathbb{R}}$ be the set of computable real numbers.

2. A real number $x \in \mathbb{R}$ is *polynomial-time computable* iff there is a Deterministic Polytime Turing Machine which on input $1^n$ computes a binary representation of an integer $m \in \mathbb{Z}$ such that $|m/2^n - x| \leq 1/2^n$. Let $\mathcal{P}_\mathbb{R}$ be the set of polynomial time real numbers.
Basic Definitions

1. A complex number $z = x + iy$ is *computable* iff $x, y \in \tilde{\mathbb{R}}$. Let $\tilde{\mathbb{C}}$ be the set of computable complex numbers.

2. A complex number $z = x + iy$ is *polynomial-time computable* iff $x, y \in \mathbb{P}_R$. Let $\mathbb{P}_C$ be the set of polynomial time computable complex numbers.

3. A normalized vector $\phi$ in any Hilbert space $\ell^2(S)$ is *computable* (*polynomial computable*) if the range of $\phi$ (a function from $S$ to complex numbers) is $\tilde{\mathbb{C}}$ ($\mathbb{P}_C$).
Outline

1. Introduction
2. The Quantum Turing Machine: Formal Description
3. The Computational Power of QTM
4. Equivalence Results on Quantum Computational Models
5. Bibliography
QTM: General

Definition

A QTM is a triplet \((\Sigma, Q, \delta)\), where:

- \(\Sigma\) is an alphabet (with an identified symbol \#);
- \(Q\) is a set of states (with \(q_0\) initial state, \(q_f\) final state, \(q_0 \neq q_f\));
- \(\delta\) is the transition function

\[
\delta: Q \times \Sigma \rightarrow \tilde{C} \times \Sigma \times \{L,R\}
\]

- The QTM has a two-way infinite tape of cells indexed by \(\mathbb{Z}\) and a read-write tape head that moves along the tapes.
Let $M$ be a QTM and let $S$ be the inner product space of finite complex linear combination of $M$ with the Euclidean norm. We call each element $\phi \in S$ a **superposition** of $M$.

**Definition**

A QTM $M$ defines a linear operator $U_M: S \rightarrow U_M$ called the time evolution operator in the following way: if $M$ starts in configuration $C$ with current state $p$ and scanned symbol $\sigma$, then after one step $M$ will be in a superposition $\psi = \sum_i \alpha_i c_i$, where each non-zero $\alpha_i$ correspond a transition $\delta(p, \sigma, \tau, q, d)$, and $c_i$ is the new configuration obtained by applying the transition to $c$. 
QTM: The Time Evolution Operator

Note:

- the set $C$ of configurations of $M$ is an orthonormal basis for $S$;
- each superposition $\psi \in S$ can be represented as a vector of complex numbers indexed by configurations;
- $U_M$ can be represented as a square matrix with columns and rows indexed by configurations where the matrix element from a column $c$ and a row $c'$ gives the amplitudes with which configuration $c$ leads to configuration $c'$ in a single step of $M$;

Definition

We say that a QTM $M$ is **well-formed** if the time evolution operator $U_M$ is unitary.
QTM: computation as unitary transformation

- QTM is obviously reversible.
- An efficient QTM implements any given unitary transformation, approximating it by a product of simple unitary transformations.
- The “super-power’ of quantum computation: reversibility, quantum parallelism, and interference of computational paths.
- It is possible to define an Universal QTM.
Observation of QTM

When a QTM $M$ in superposition $\psi = \sum_i \alpha_i c_i$ is observed or measured, a configuration $c_i$ is observed with probability $\alpha_i$. Moreover, the superposition of $m$ is updated to $\psi' = c_i$.

It is also possible to perform a partial measurement; for example, suppose we want to observe the first cell (which contains 0 or 1). Suppose the superposition is $\psi = \sum_i \alpha_0 i c_0 i + \psi = \sum_i \alpha 1 i c 1 i$. If 0 is observed, $\Pr[0] = \psi = \sum_i |\alpha_0 i|^2$ and the new superposition is given by $1 / \sqrt{\Pr[0]} \psi = \sum_i \alpha_0 i c_0 i$.

In general, the output of a QTM is a sample from a probability distribution.
Quantum Turing machines need some input/output conventions.

**Definition**

- We consider *final configuration* any configuration in a QTM $M$ in the final state $q_f$.
- We say that a QTM $M$ *halts with running time* $T$ on input $x$ if when $M$ is run with input $x$, at time $T$ the superposition contains only final configurations, and at any times $T_i < T$ the superposition contains no final configurations.

Bernstein and Vazirani in [1] give also careful definitions on the output of QTM.
Definition (Stationarity and Normal Form)

- A QTM $M$ is called *well behaved* if it halts on all input strings in a final superposition where each configuration has the tape head in the same cell.
- If this cell is always the start cell, we call $M$ *stationary*.
- We say that $M$ is in *normal form* if it is well formed and $q_f$ always leads back to $q_0$.

Definition (Unidirectionality)

A QTM is called *unidirectional* if each state can be entered from only one direction.
Definition (Multitrack TM)

A multitrack TM with $k$ tracks is a TM whose alphabet $\Sigma$ is of the form $\Sigma_1 \times \ldots \times \Sigma_k$ with a special blank symbol $\#$ in each $\Sigma_i$ such that the blank in $\Sigma$ is $(\#, \ldots, \#)$. The input is specified by specifying the string on each track. So the TM on input $x_1; \ldots; x_k \in \prod_{i=1}^k (\Sigma_i - \#)$ is started in the initial configuration with the non-blank portion of the i-th coordinate of the tape containing the string $x_i$ starting in the start cell.
How we can verify that the QTM $M$ effectively halts? Bernstein and Vazirani in [1] write: "This can be accomplished by performing a measurement to check whether the machine is in the final state $q_f$. Making this partial measurement does not have any other effect on the computation".

This can be "implemented" with an observation cell in which we can perform a partial measurement.
Outline

1. Introduction
2. The Quantum Turing Machine: Formal Description
3. The Computational Power of QTM
4. Equivalence Results on Quantum Computational Models
5. Bibliography
Accepting languages with QTM

Definition

Let $M$ a stationary, normal form, multitrack QTM whose last track has alphabet $\{\#, 0, 1\}$. If we run $M$ with string $x$ in the first track and the empty string elsewhere, wait until $M$ halts and then observe the last track of the start cell: we will see a 1 with probability $p$. We will say that $M$ accepts $x$ with probability $p$ and rejects $x$ with probability $1 - p$.

Definition

We say that a QTM $M$ accepts a language $\mathcal{L}$ with probability $p$, if $M$ accepts with probability at least $p$ every string $x \in \mathcal{L}$, and rejects with probability at least $p$ every string $x \notin \mathcal{L}$.
Quantum Complexity Classes

**Definition**

The class $EQP$ is the set of the languages $\mathcal{L}$ accepted by polynomial QTM $M$ with probability 1.

$EQP$ is the error-free (or exact) quantum polynomial-time complexity classes.

**Definition**

The class $BQP$ is the set of the languages $\mathcal{L}$ accepted by polynomial QTM $M$ with probability 2/3.
Quantum Complexity Classes

**Definition**

The class $ZQP$ is the set of the languages $\mathcal{L}$ accepted by polynomial QTM $M$ such that, for every string $x$:

- if $x \in \mathcal{L}$, then $M$ accepts $x$ with probability $p > 2/3$ and rejects with probability $p = 0$;
- if $x \notin \mathcal{L}$, then $M$ rejects $x$ with probability $p > 2/3$ and accepts with probability $p = 0$.

The class $ZQP$ is the zero-error extension of the class $BQP$. In fact the QTM never gives a wrong answer, but in each case with probability 1/3 gives a “don’t-know” answer (clearly, in this case we need to have three answers).
Quantum Complexity Classes

Some results:

- The inclusions $EQP \subseteq ZQP \subseteq BQP$ obviously hold.
- The relationship with classical complexity classes is the following:

$$P \subseteq BPP \subseteq BQP \subseteq \text{PSPACE}$$
Outline

1. Introduction

2. The Quantum Turing Machine: Formal Description

3. The Computational Power of QTM

4. Equivalence Results on Quantum Computational Models

5. Bibliography
QTM and Quantum Circuit Families

On total computations, the most relevant quantum computational models, i.e. QTM and Quantum Circuit Families are equivalent.

In [4] Yao propose an interesting encoding of the QTM in terms of Quantum Circuit Families.
The quantum circuit computing one step of the simulation.
The interesting case of polynomial time quantum computations has been largely investigate by Nishimura and Ozawa [3]. It is possible to define a “perfect equivalence” result between polynomial time QTM and a particular class of Quantum Circuit Families.

**Definition (Polynomial-Time QTM)**

A *polynomial time* QTM $M$ is a QTM which on every input $x$ halts in time $T$ with $T$ polynomial in the length of $x$. 
The Perfect Equivalence

The perfect equivalence needs some hypothesis:

- Amplitudes of the QTM $M$ have to be in $\mathbb{P}C$.
- We restrict Quantum Circuit Families to the sub-class of the *Finitely Generated* one.
Outline

1. Introduction
2. The Quantum Turing Machine: Formal Description
3. The Computational Power of QTM
4. Equivalence Results on Quantum Computational Models
5. Bibliography

