Journal of Universal Computer Science, vol. 32, no. 3 (2026), 357-372
submitted: 25/4/2025, accepted: 5/9/2025, appeared: 28/3/2026

State transition diagrams for a universal quantum gate set

Ayodhya Liyanage
(Department of Mathematics, University of Colombo, Colombo 03, Sri Lanka
https://orcid.org/0009-0009-6115-3936, 1224liyanage@gmail.com)

Anuradha Mahasinghe
(Department of Mathematics, University of Colombo, Colombo 03, Sri Lanka
https://orcid.org/0000-0003-2828-6090, anuradhamahasinghe@maths.cmb.ac.lk)

Abstract: A quantum Turing machine (QTM) is an abstract model of computing that serves as the quantum counterpart of a classical Turing machine. Closely related to the probabilistic Turing machine, a QTM utilises quantum effects such as superposition, entanglement and unitary evolution. Despite its historical role as a framework for devising quantum algorithms and the significance of its classical counterpart in theoretical computing, very little attention has been paid in literature to the state transition of QTM's for basic quantum gates. In this paper, we construct the state transition diagrams for a set of elementary quantum gates that consists of the Hadamard, CNOT, and T gates, providing a universal basis for fault tolerant quantum computation. We verify the necessary conditions to ensure that the designed state transition diagrams comply with the postulates of quantum mechanics, verifying their well formedness.

Keywords: Quantum computing, Models of computation, Turing machines, Quantum gates, Quantum algorithms
Categories: F.1.1, F.1.m, G.2.3, G.2.0
DOI: 10.3897/jucs.156960

1 Introduction

In 1980, Paul Benioff developed a quantum version of Alan Turing's abstract computing machine [Benioff 1980], demonstrating the possibility of making a *quantum computer* – a computational device that takes into account quantum effects. Shortly afterwards, Richard Feynman suggested that quantum computers could outperform classical computers, and pointed out that a probabilistic Turing machine cannot simulate a general quantum mechanical system without an exponential slowdown [Feynman 1982]. In light of Feynman's observation, researchers began investigating the computability and complexity of a computing device based on quantum effects.

In 1985, David Deutsch introduced a more precise abstract computing model that utilised quantum effects[Deutsch 1985], and it was known as the *quantum Turing machine* (QTM). He identified some major drawbacks in Benioff's model and proved that classical Turing machines could efficiently simulate every computation in Benioff's model. Further, Deutsch's famous quantum algorithm for determining whether a given promise function is constant or balanced, was described within the QTM model. By re–examining the foundations of computational complexity theory, E. Bernstein and U. Vazirani proved the existence of a universal quantum Turing machine with polynomially bounded simulation overhead and provided more rigorous definitions of QTM's [Bernstein and Vazirani 1993, Bernstein and Vazirani 1997].

Thus, since its inception, quantum computing has benefited significantly from the QTM model. Further, in 1993 Yao proved the polynomial equivalence between QTM and quantum circuit models, implying any quantum circuit can be simulated by a QTM with a polynomial overhead [Yao 1993]. However, it was the circuit model that provided the framework to devise almost all major quantum algorithms subsequent to Deutsch's algorithm. Consequently, the QTM model received less attention during the rapid growth of quantum computing. Therefore, unlike in classical computing, expressing computational tasks in the form of state transition diagrams has not been the case with quantum computing. Having said that, one distinguishable work by Liang and Yang appeared in 2013 [Liang and Yang 2013], presenting state transition diagrams for several quantum gates including Hadamard, CNOT and Toffoli, developed upon a variant of QTM's. However, the main focus of this work has been exploring quantum computers' capability of addressing the halting problem; which has possibly led the authors to define their variant of abstract machines not aligning with quantum postulates such as unitarity. Consequently, their version of QTM does not obey the *well–posedness* required by the QTM definitions by Deutsch, Bernstein and Vazirani. In this context, designing state transition diagrams for basic quantum operations in the standard QTM model, preserving well–posedness, is an unexplored topic to date. Furthermore, due to the necessary conditions on well–posedness in the standard QTM model, it becomes a non–trivial task as well.

In this paper we fill this gap by presenting well–posed state transition diagrams based on the standard QTM model for a set of basic quantum gates: Hadamard, CNOT, and T. These gates were proven to make a universal fault–tolerant basis for quantum computation [Boykin et al. 2000]. Therefore, our transition diagrams can be used to simulate any given quantum computation in the circuit–gate model to a desired accuracy.

The paper is organised as follows: we provide basic definitions and theorems in section 2. Following the previous works by Bernstein and Vazirani, we state the necessary conditions for a well–formed QTM in section 3. Our main contributions – the transition diagrams with proofs are presented in section 4. Concluding remarks are included in section 5.

2 Basic definitions

2.1 Probabilistic Turing machines

A Probabilistic Turing Machine (PTM) is a type of nondeterministic TM that chooses the next transition according to some probability distribution at each computational step, formally defined as follows:

Definition 2.1. A PTM is a triplet (Q, Σ, δ) where Q is a finite set of states including q_0, q_{accept} where $q_0 \neq q_{accept}$ and Σ is the finite alphabet with blank symbol # and δ is the state transition function,

$$\delta : Q \times \Sigma \times \Sigma \times Q \times \{L, R\} \to [0, 1].$$

The notation $\delta(p, \sigma, \gamma, q, d)$ denotes the probability that the machine in state p and reads the symbol σ will change the symbol to γ, shift to state q and move one cell in the direction d [Bernstein and Vazirani 1993].

The state transition function of a PTM must satisfy the following condition to ensure that the machine chooses one transition based on a probability distribution over all possible transitions: for all $(p, \sigma) \in \Sigma \times Q$, $\sum_{\gamma,q,d} \delta(p, \sigma, \gamma, q, d) = 1$ [Santos 1969].

In a PTM, the state transition function δ can be represented as a stochastic square matrix M in the space of configuration, where a_{ij} corresponds to the element in row i and column j. More specifically, $a_{ij} = Pr[c_j \to c_i] \in [0,1]$, representing the probability of transition from configuration c_j to configuration c_i.

2.2 Quantum Turing machines

Recall the QTM is an abstract computational model designed to probe the edges of computability and complexity within the realm of quantum computing, for a set of formal definitions related to QTM's, we adopt the following from Bernstine and Vazirani [Bernstein and Vazirani 1997].

Definition 2.2. $\tilde{\mathbb{C}}$ is the set of all $\alpha \in \mathbb{C}$, for which there exists an algorithm to compute the real and imaginary parts of α runs in polynomial time in n within the error of at most 2^{-n}.

Definition 2.3. A QTM is formally defined by a triplet (Σ, Q, δ), where Σ is the finite alphabet that includes the blank symbol #, Q is the finite set of states with a specified initial state q_0 and final state $q_f \neq q_0$, and δ is the quantum state transition function,

$$\delta : Q \times \Sigma \to \tilde{\mathbb{C}}^{\Sigma \times Q \times \{L,R\}}$$

QTM is also equipped with a two-way infinite tape indexed by $\mathbb{Z}$ and a tape head that allows one to read, write, and move along the tape [Bernstein and Vazirani 1997].

Similarly to the classical version of Turing machines, the configuration of a QTM is also defined in the same manner. However, unlike classical TMs, QTMs have the ability to exist in a linear combination of configurations at a single point in time by reflecting the inherent quantum physical nature of *superposition*.

Definition 2.4. [Bernstein and Vazirani 1993] When a QTM is in a superposition of configurations $\sum \alpha_i c_i$ and is measured, the probability of observing the configuration c_i is $|\alpha|^2$ and the superposition $\sum \alpha_i c_i$ collapses to c_i.

The above definition describes another effect of quantum mechanics in QTM known as quantum *measurement*. In addition, this may perform partial measurements in QTMs by measuring only some of the cells, not all.

2.3 Quantum state transition function

The state transition function δ is defined as a relationship between the current configurations of the machine and the following configurations in response to an event. If the machine is in state p and the head of the tape reads symbol σ, the function $\delta(p,\sigma)$ produces a complex vector that belongs to a complex vector space of dimension k, indexed by $(\gamma, q, d) \in \Sigma \times Q \times \{L,R\}$ where $k = |\Sigma \times Q \times \{L,R\}|$. The amplitude determines the operation in which the tape head replaces σ with γ' and transitions the machine to the state q' by moving the tape head one cell to the direction d given by the value corresponding to the row indexed with (γ', q', d) in $\delta(p,\sigma)$.

The preceding definition of the quantum state transition function can be refined to simplify expressing the amplitude for a specific transition [Bernstein and Vazirani 1993].

$$\delta : Q \times \Sigma \times \Sigma \times Q \times \{L, R\} \to \tilde{\mathbb{C}}$$

Similarly to probabilistic Turing machines, $\delta(p, \sigma, \gamma, q, d) \in \tilde{\mathbb{C}}$ gives the complex amplitude of the transition when the machine is in state p and reads the symbol σ, then replaces the symbol σ with γ, transitions to state q, and moves one cell in the direction d.For convenience, we can denote the amplitude α of the transition $\delta(p, \sigma, \gamma, q, R)$ as $\alpha \left|\gamma\right\rangle \left|q\right\rangle \left|d\right\rangle$ [Bernstein and Vazirani 1997]. Here, $\left|\gamma\right\rangle \left|q\right\rangle \left|d\right\rangle$ corresponds to the vector produced by $\delta(p, q)$, where the entry indexed by $(\gamma.q, d)$ is one, and all other elements remain zero.

2.4 Time evolution operator

Let M be a QTM. Consider an inner product space over $\mathbb{C}$ of n dimensions, where n represents the number of distinct configurations in M. For each i such that $1 \leq i \leq n$, let $\left|c_i\right\rangle$ represent a unit vector in S with a value of 1 in the i^{th} position and 0 elsewhere. For any two distinct vectors $\left|c_i\right\rangle$ and $\left|c_j\right\rangle$ the inner product satisfies $\langle c_i | c_j \rangle = 0$. Since the set of vectors $\{\left|c_i\right\rangle \mid 1 \leq i \leq n\}$ consists of mutually orthogonal unit vectors, it serves as an orthonormal basis for S. With regard to this standard basis. Consequently, S is an inner product space of complex linear combinations of configurations of M with the Euclidean norm, and every element $\psi \in S$ represents a superposition of configurations of M. Then, QTM M defines a linear operator $U_M : S \to S$ called *time evolution operator* of QTM M, which dictates the transformations of the QTM state in a single time step[Bernstein and Vazirani 1993]. For M begins with the configuration c, where it is in state p and scans the symbol σ after a single time step, M evolves into a superposition of the configurations given by $\psi = \sum \alpha_i c_i$. Here, α_i denotes the nonzero amplitudes corresponding to the transition $\delta(p, \sigma)$, and c_i is the new configuration resulting from applying this transition to c.

We can define the time evolution operator U_M for QTM M by a countable dimensional square matrix, rows, and columns indexed by the configurations of M [Bernstein and Vazirani 1997]. The element corresponds to row c, and column c' represents the transition amplitude, where configuration c' transits to configuration c in a single time step.

2.5 Variants of QTM's

When analysing the behaviour of a QTM, it is possible to categorise it into different types. All these variants are governed by the above definitions and conventions, with some significant variations. We introduce some of the important variants in QTM that will be useful in future advancements.

2.5.1 Unidirectional Quantum Turing Machines

Definition 2.5. [Bernstein and Vazirani 1997] A QTM =(Q, Σ, δ) is called unidirectional if each state in Q can be entered only from one direction.

The presence of transitions $\delta(p_1, \sigma_1, \gamma_1, q, d_1)$ and $\delta(p_2, \sigma_2, \gamma_2, q, d_2)$ with nonzero amplitudes within a unidirectional QTM implies $d_1 = d_2$. Although not all QTMs possess unidirectionality, it has been established that a unidirectional QTM is capable of simulating any QTM efficiently.

2.5.2 Normal form Quantum Turing Machines

Definition 2.6. [Bernstein and Vazirani 1997] A QTM = (Q, Σ, δ) is in normal form if, for all $\sigma \in \Sigma$,

$$\delta(q_f, \sigma) = |q_0\rangle\,|\sigma\rangle\,|R\rangle$$

where $q_f \in Q$ is the final state.

This implies that without changing its symbol, every transition in the final state moves in the right (R) direction to the initial state q_0. Such behaviour will be beneficial in defining a well-formed QTM, as all configurations in the final state are redirected to the initial state, allowing a transition to the initial configurations without violating reversibility.

3 Well-formedness of a QTM

Definition 3.1. [Bernstein and Vazirani 1993] For QTM M, we said that M is well formed if its time evolution operator U_M preserves the Euclidean norm.

Well-formedness is a necessary condition for QTM to be consistent with quantum physics, which implies that for every superposition of QTM in consecutive steps, the sum of the probabilities of all possible configurations is equal to one. Upon further examination, it becomes apparent that the unitary time evolution, which is necessary for upholding the fundamental principles of quantum physics, is equivalent to the well-formedness of the Quantum Turing Machine (QTM) defined above. Therefore, the following theorem can be formulated.

Theorem 1. *[Bernstein and Vazirani 1993] A QTM is well formed if and only if its time evolution operator is unitary.*

Unitarity, a fundamental principle in quantum mechanics, is essential to ensure that the evolution of a closed quantum system remains reversible and preserves probability. To uphold this principle, the time evolution operator must be unitary. Mathematically, the time evolution of a quantum state $|\psi(t)\rangle$ is described by the schrödinger equation. The solution to the schrödinger equation leads to the time evolution operator, given by,

$$U(t, t_0) = e^{-iH(t-t_0)/\hbar}$$

which evolves the initial state $|\psi(t_0)\rangle$ to a new state at time t as $|\psi(t)\rangle = U(t, t_0)\,|\psi(t_0)\rangle$ and by the unitarity of the time evolution operator is satisfied that $U^\dagger U = UU^\dagger = I$ where $U^\dagger$ is the conjugate transpose of U [Nielsen and Chuang 2010]. Since the linear operator U_M, known as the time evolution operator of QTM M, governs the transformations from one quantum state to another in a single time step, ensuring that U_M is unitary, we guarantee that the evolution of QTM follows the fundamental principle of unitary time evolution from quantum mechanics.
To ensure the unitarity of the time evolution operator U, it is necessary to show $UU^\dagger = U^\dagger U = I$. Unfortunately, it is computationally intensive to check this condition, especially for complex quantum systems. Therefore, an alternative method is required to verify the unitarity of the time evolution operator.

Theorem 2. *[Bernstein and Vazirani 1997] A QTM M defined by (Q, Σ, δ) is well-formed if and only if the following conditions hold:*

1. *For all $(p, \sigma) \in Q \times \Sigma$,*
$$\|\delta(p, \sigma)\| = 1.$$
2. *For all $(p_1, \sigma_1) \neq (p_2, \sigma_2) \in Q \times \Sigma$,*
$$\delta(p_1, \sigma_1) \cdot \delta(p_2, \sigma_2) = 0.$$
3. *For all $(p_1, \sigma_1, \gamma_1), (p_2, \sigma_2, \gamma_2) \in Q \times \Sigma \times \Sigma$,*
$$\delta(p_1, \sigma_1 \mid \gamma_1, R) \cdot \delta(p_2, \sigma_2 \mid \gamma_2, L) = 0.$$

The restriction of the transition function $\delta(p, \sigma)$ that transforms from the tape symbol σ to γ and moves in the direction d given by $\delta(p, \sigma|\gamma, d)$. This can be represented by a $|Q|$ - dimensional complex vector indexed by the state of the finite state set.

According to the properties of a unitary matrix, a matrix U is unitary if and only if the columns of U are mutually orthogonal and have unit length. Applying three criteria of the above theorem, we can prove the unitarity of the time evolution operator U by showing that all columns are orthonormal. Consider a QTM M with a time evolution operator U_M.The first condition specifies the unit length requirement that each column of U_M has exactly a unit length. The orthogonality and separability conditions, stated as the second and third, confirmed the orthogonality between any two columns of U_M.In the context of unidirectional QTMs, only the first two conditions of Theorem 2 are sufficient to prove that the relevant QTM is well formed [Bernstein and Vazirani 1997].

The study of Liang and Yang [Liang and Yang 2013] on stationary rotational QTM (SR-QTM), designed to address the halting scheme problem on the class of QTMs, made a significant effort to formulate basic quantum gates for SR-QTMs.The stationary rotational QTM is a subclass of QTMs that inherits stationary and rotational behaviour [Bernstein and Vazirani 1997]. The study employed three basic quantum gates: the Hadamard gate, which allows for quantum superposition of a single qubit, and the multi-qubit C-NOT and Toffoli gates. In accordance with their convention, they define their variant of a QTM using a one-way infinite tape instead of a two-way tape, and for all inputs, the machine starts from the leftmost cell, indexed as 0, while the 0^{th} cell is always filled with the blank symbol #. Analysing these SR-QTMs shows that their transition functions are not well defined since for certain states in Q, the transition function δ is not defined for all $\sigma \in \Sigma$, which results in a partially defined QTM. Although partial definitions are acceptable in classical Turing machines, they create fundamental issues in the quantum version.

By studying these SR-QTMs, we observe that the machine uses only a finite portion of the tape, which allows us to bound the space of configurations to a finite set. Consequently, it is possible to represent the time evolution operator as a finite square matrix [Bernstein and Vazirani 1997]. However, because of the partially defined state transition function, some configurations cannot be reached by any transition. As a result, the corresponding columns of the matrix are not orthonormal, violating the requirement for the matrix to be unitary.

4 State transition diagrams

4.1 Quantum state transition

The idea of a Quantum State Transition Diagram (QSTD) was introduced by [Hook and Lee 2010] as a formal tool to represent the quantum cellular automata for some quantum operations. However, due to the greater computational expressiveness and unbounded operational capacity which are not shared with quantum cellular automata, Turing machines necessitate a more advanced and formally rigorous representational framework. By refining this conceptual model, Liang and Yang introduced a new version of QSTD for use with QTMs [Liang and Yang 2013]. However, when investigating the Liang and Yang version of QSTD, because the way indicates the direction of the tape head's movement, this version is only capable of illustrating unidirectional QTMs. Therefore, we recognised the importance of a generalised method for representing all types of quantum Turing machines, and here we introduced an improved version of QSTD.

Using the improved QSTD framework, a single transition of $\delta(p, \sigma, \gamma, q, R) = \alpha$ can be represented as follows.

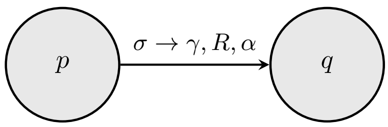

Figure 1: QSTD description of the transition $\delta(p, \sigma, \gamma, q, R) = \alpha$

As shown in the figure 1 above, this improved version of QSTD can represent all types of QTM without restrictions. The edge between node p and node q, $\sigma \to \gamma, R, \alpha$ denotes the transition between state p to state q where the symbol σ replaces γ and moves in the right direction (R) with the complex amplitude α.

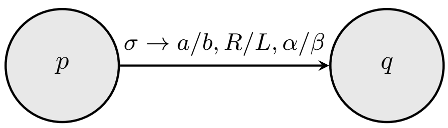

Figure 2: QSTD description of the transition $\delta(p, \sigma) \to \alpha |a\rangle |q\rangle |R\rangle + \beta |b\rangle |q\rangle |L\rangle$

Figure 2 illustrates the QSTD of a Quantum Turing Machine transition from a single configuration to a superposition of configurations.

4.2 Hadamard gate

Theorem 3. *There is a well-formed normal form QTM that takes the initial superposition of n-bit strings $|\psi\rangle$ and produces the final superposition of $H_n |\psi\rangle$.*

Let QTM M $=(Q, \Sigma, \delta)$ where alphabet $\Sigma \in \{\#, 0, 1\}$ and finite state set $Q \in \{q_0, q_1, q_2, q_3, q_4, q_f\}$ with the quantum state transition function δ define as follows:

$$\delta(q_0, \#) \to |\#\rangle\,|q_1\rangle\,|R\rangle \qquad \delta(q_3, \#) \to |\#\rangle\,|q_4\rangle\,|R\rangle$$
$$\delta(q_0, 0) \to |0\rangle\,|q_1\rangle\,|R\rangle \qquad \delta(q_3, 0) \to |0\rangle\,|q_3\rangle\,|L\rangle$$
$$\delta(q_0, 1) \to |1\rangle\,|q_1\rangle\,|R\rangle \qquad \delta(q_3, 1) \to |1\rangle\,|q_3\rangle\,|L\rangle$$
$$\delta(q_1, \#) \to |\#\rangle\,|q_2\rangle\,|L\rangle \qquad \delta(q_4, \#) \to |\#\rangle\,|q_f\rangle\,|L\rangle$$
$$\delta(q_1, 0) \to |0\rangle\,|q_4\rangle\,|R\rangle \qquad \delta(q_4, 0) \to |0\rangle\,|q_f\rangle\,|L\rangle$$
$$\delta(q_1, 1) \to |1\rangle\,|q_4\rangle\,|R\rangle \qquad \delta(q_4, 1) \to |1\rangle\,|q_f\rangle\,|L\rangle$$
$$\delta(q_2, \#) \to |\#\rangle\,|q_3\rangle\,|R\rangle \qquad \delta(q_f, \#) \to |\#\rangle\,|q_0\rangle\,|R\rangle$$
$$\delta(q_2, 0) \to \frac{1}{\sqrt{2}}\,|0\rangle\,|q_2\rangle\,|R\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\,|q_2\rangle\,|R\rangle \qquad \delta(q_f, 0) \to |0\rangle\,|q_0\rangle\,|R\rangle$$
$$\delta(q_2, 1) \to \frac{1}{\sqrt{2}}\,|0\rangle\,|q_2\rangle\,|R\rangle - \frac{1}{\sqrt{2}}\,|1\rangle\,|q_2\rangle\,|R\rangle \qquad \delta(q_f, 1) \to |1\rangle\,|q_0\rangle\,|R\rangle$$

For an initial configuration consisting of an input $x \in (\Sigma - \{\#\})^*$ with a length of n organised in cells indexed from 0 to $n-1$, the machine will operate as follows. Computation starts in the initial state q_0 with a tape head on the cell index 0. After the machine moves right and left, it enters q_2 and moves right along the symbols by performing the Hadamard transition,

$$\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}$$

to each qubit, where an equal superposition of the two basis states is created. After reaching the symbol #, the machine identifies the right bound of the input and the system transitions to the state q_3 by moving to the left until it encounters the leftmost #. The machine then enters state q_4 and moves to the right, reverses direction to the left, enters the final state q_f, and halts with the tape head positioned above the cell indexed at -1. By the definition 2.6 and transition defined for q_f in δ, QTM M appeared to be in normal form. Moreover, for all $p, \sigma \in Q \times \Sigma$, $\delta(p, \sigma)$ has unit length, indicating that every transition in this QTM preserves the Euclidean norm. In addition, the inner product of two different $\delta(p_1, \sigma_1) \neq \delta(p_2, \sigma_2) \in Q \times \Sigma$ is also found to be zero. Thus, unit length and orthogonality properties combined with the unidirectional nature, according to Theorem 2 QTM M is well formed.

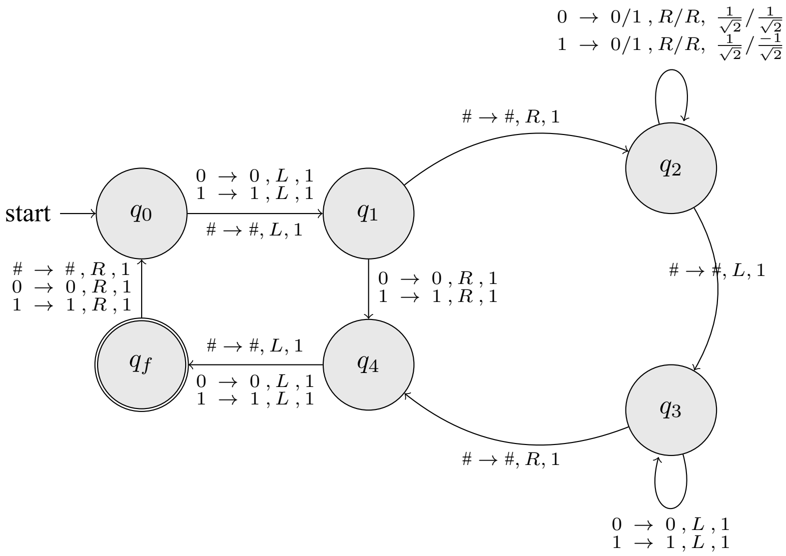

Figure 3: Quantum state transition diagram for the well-defined, normal form QTM executing the Hadamard transformation for n-qubits.

According to the transitions defined within the above transition function δ, a quantum state transition diagram can be constructed as a graphical representation of the QTM that performs a Hadamard transformation of n bits as in Figure 3.

We shall examine the computational mechanism of the above QTM, which executes a Hadamard operation on n qubits. According to the definition, every computation of a QTM must initiate from a classical configuration, similar to deterministic Turing machines [Molina and Watrous 2019]. But to investigate computability, here we perform the Hadamard transition for the input of a single qubit in a superposition $a\,|0\rangle + b\,|1\rangle$, where $|a|^2 + |b|^2 = 1$.

$$\xrightarrow{0} a\,|q_0\rangle\,|\#0\#\rangle\,|0\rangle + b\,|q_0\rangle\,|\#1\#\rangle\,|0\rangle$$

$$\xrightarrow{1} a\,|q_1\rangle\,|\#0\#\rangle\,|-1\rangle + b\,|q_1\rangle\,|\#1\#\rangle\,|-1\rangle$$

$$\xrightarrow{2} a\,|q_2\rangle\,|\#0\#\rangle\,|0\rangle + b\,|q_0\rangle\,|\#1\#\rangle\,|0\rangle$$

$$\xrightarrow{3} a\Big(\frac{1}{\sqrt{2}}\,|q_2\rangle\,|\#0\#\rangle\,|1\rangle + \frac{1}{\sqrt{2}}\,|q_2\rangle\,|\#1\#\rangle\,|1\rangle\Big) + b\Big(\frac{1}{\sqrt{2}}\,|q_2\rangle\,|\#0\#\rangle\,|1\rangle - \frac{1}{\sqrt{2}}\,|q_2\rangle\,|\#1\#\rangle\,|1\rangle\Big)$$

$$= \frac{a+b}{\sqrt{2}}\,|q_2\rangle\,|\#0\#\rangle\,|1\rangle + \frac{a-b}{\sqrt{2}}\,|q_2\rangle\,|\#1\#\rangle\,|1\rangle$$

$$\xrightarrow{4} \frac{a+b}{\sqrt{2}}\,|q_3\rangle\,|\#0\#\rangle\,|0\rangle + \frac{a-b}{\sqrt{2}}\,|q_3\rangle\,|\#1\#\rangle\,|0\rangle$$

$$\xrightarrow{5} \frac{a+b}{\sqrt{2}}\,|q_3\rangle\,|\#0\#\rangle\,|-1\rangle + \frac{a-b}{\sqrt{2}}\,|q_3\rangle\,|\#1\#\rangle\,|-1\rangle$$

$$\xrightarrow{6} \frac{a+b}{\sqrt{2}} |q_4\rangle |\#0\#\rangle |0\rangle + \frac{a-b}{\sqrt{2}} |q_4\rangle |\#1\#\rangle |0\rangle$$
$$\xrightarrow{7} \frac{a+b}{\sqrt{2}} |q_f\rangle |\#0\#\rangle |-1\rangle + \frac{a-b}{\sqrt{2}} |q_f\rangle |\#1\#\rangle |-1\rangle$$

As defined, the input should be placed from the 0^{th} cell, and all other cells are blank. A quantum configuration can be represented by three quantum states denoted by the three ket notation. The first quantum state represents the current state of the machine, and the second represents the content of the tape. Since the QTM tape is two-way infinite, we only denote the finite portion of the tape that will be used for the relevant computation. In this case, we only denote the cells from -1 to 1, which contain the input and the leftmost and rightmost #. The last state represents the current position of the tape head. The initial configuration for this computation represents $|q_0\rangle |\#0\#\rangle |0\rangle$ or $|q_0\rangle |\#1\#\rangle |0\rangle$ because the initial configuration should not be in a superposition. We can see how the Hadamard operation applies to the qubits in the q_2 state. Ultimately, the machine enters the final state q_f and halts the computation.

4.3 Controlled NOT gate

To perform the Controlled NOT (C-NOT) operation, we must create a QTM that leaves the qubit state unchanged for the inputs $|00\rangle$ and $|01\rangle$, while transforming $|10\rangle$ into $|11\rangle$ and $|11\rangle$ to $|10\rangle$, respectively, for the input states $|10\rangle$ and $|11\rangle$.

Theorem 4. *There exists a well-formed, normal-form QTM that operates on a quantum register consisting of two qubits, with the first qubit acting as the control qubit and the other as the target qubit.*

We can construct such a machine using $\Sigma = \{0, 1, \#\}$, the finite state set $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_f\}$ and the state transition function δ which defines the transition rules governing the behaviour of the QTM.

$$\begin{array}{lll}
\delta(q_0,\#) \rightarrow |\#\rangle |q_1\rangle |L\rangle, & \delta(q_0,0) \rightarrow |0\rangle |q_1\rangle |L\rangle, & \delta(q_0,1) \rightarrow |1\rangle |q_1\rangle |L\rangle \\
\delta(q_1,\#) \rightarrow |\#\rangle |q_2\rangle |R\rangle, & \delta(q_1,0) \rightarrow |0\rangle |q_8\rangle |R\rangle, & \delta(q_1,1) \rightarrow |1\rangle |q_8\rangle |R\rangle \\
\delta(q_2,\#) \rightarrow |\#\rangle |q_3\rangle |R\rangle, & \delta(q_2,0) \rightarrow |0\rangle |q_3\rangle |R\rangle, & \delta(q_2,1) \rightarrow |1\rangle |q_4\rangle |R\rangle \\
\delta(q_3,\#) \rightarrow |\#\rangle |q_5\rangle |L\rangle, & \delta(q_3,0) \rightarrow |0\rangle |q_5\rangle |L\rangle, & \delta(q_3,1) \rightarrow |1\rangle |q_5\rangle |L\rangle \\
\delta(q_4,\#) \rightarrow |\#\rangle |q_6\rangle |L\rangle, & \delta(q_4,0) \rightarrow |1\rangle |q_6\rangle |L\rangle, & \delta(q_4,1) \rightarrow |0\rangle |q_6\rangle |L\rangle \\
\delta(q_5,\#) \rightarrow |\#\rangle |q_7\rangle |L\rangle, & \delta(q_5,0) \rightarrow |0\rangle |q_7\rangle |L\rangle, & \delta(q_5,1) \rightarrow |1\rangle |q_3\rangle |R\rangle \\
\delta(q_6,\#) \rightarrow |\#\rangle |q_4\rangle |R\rangle, & \delta(q_6,0) \rightarrow |0\rangle |q_4\rangle |R\rangle, & \delta(q_6,1) \rightarrow |1\rangle |q_7\rangle |L\rangle \\
\delta(q_7,\#) \rightarrow |\#\rangle |q_8\rangle |R\rangle, & \delta(q_7,0) \rightarrow |0\rangle |q_2\rangle |R\rangle, & \delta(q_7,1) \rightarrow |1\rangle |q_2\rangle |R\rangle \\
\delta(q_8,\#) \rightarrow |\#\rangle |q_f\rangle |L\rangle, & \delta(q_8,0) \rightarrow |0\rangle |q_f\rangle |L\rangle, & \delta(q_8,1) \rightarrow |1\rangle |q_f\rangle |L\rangle \\
\delta(q_f,\#) \rightarrow |\#\rangle |q_0\rangle |R\rangle, & \delta(q_f,0) \rightarrow |0\rangle |q_0\rangle |R\rangle, & \delta(q_f,1) \rightarrow |1\rangle |q_0\rangle |R\rangle
\end{array}$$

Since the QTM intended to perform a two-qubit operation, the input should consist of two qubits placed in the 0^{th} and 1^{st} cells as the control and target qubits, respectively, and all other cells are blank. As is customary, the machine starts from the initial state q_0 with the tape head in the 0^{th} cell. C-NOT operation can be described using the following matrix:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

In state q_0, the machine steps left and right to enter state q_2. At this point, the computation branches into two distinct paths: path 1 is implemented by states q_3 through q_7, and path 2 by q_4 through q_7, depending on the control qubit. Suppose the machine reads the control qubit as $|0\rangle$. In that case, the tape head enters the state q_3, the target qubit remains unchanged and moves along the path 1, and if the control qubit reads $|1\rangle$, the machine enters the state q_4, triggering a bit flip operation on the target qubit from $|0\rangle$ to $|1\rangle$ or vice versa and moves along the path 2.
This machine also inherits a unidirectional nature. Given that every transition from q_f returns to the starting state q_0, the system can be identified as a normal form QTM. Furthermore, by Theorem 2 and employing the unit length and orthogonality properties of the transition function, it is easy to verify that the resulting QTM is well formed.

Figure 4 illustrates the quantum state transition diagram for implementing the C-NOT gate of the QTM by depicting the state transitions defined by the transition function δ.

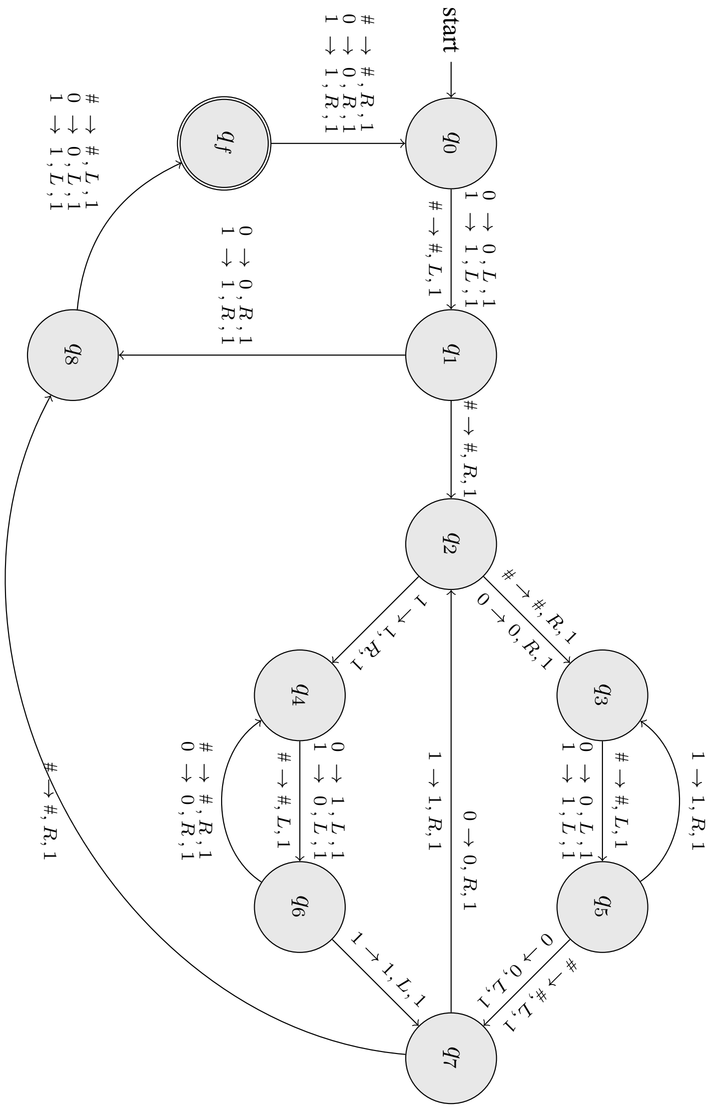

Figure 4: Quantum state transition diagram for the well-defined, normal form QTM executing the Controlled NOT gate operation.

4.4 T gate

Theorem 5. *For a single qubit state $|\psi\rangle$, a well-formed normal form QTM can be constructed to perform a phase shift of $\frac{\pi}{4}$ in the state $|\psi\rangle$.*

We define a QTM with alphabet $\Sigma \in \{\#, 0, 1\}$, a finite state set $Q = \{q_0, q_1, q_2, q_3, q_4,$

$q_f\}$ and a quantum state transition function δ that governs the transition.The following transitions of δ explicitly define the behaviour of this QTM.

$$
\begin{array}{lll}
\delta(q_0,\#) \rightarrow |\#\rangle\,|q_1\rangle\,|L\rangle, & \delta(q_0,0) \rightarrow |0\rangle\,|q_1\rangle\,|L\rangle, & \delta(q_0,1) \rightarrow |1\rangle\,|q_1\rangle\,|L\rangle, \\
\delta(q_1,\#) \rightarrow |\#\rangle\,|q_2\rangle\,|R\rangle, & \delta(q_1,0) \rightarrow |0\rangle\,|q_4\rangle\,|R\rangle, & \delta(q_1,1) \rightarrow |1\rangle\,|q_4\rangle\,|R\rangle, \\
\delta(q_2,\#) \rightarrow |\#\rangle\,|q_3\rangle\,|L\rangle, & \delta(q_2,0) \rightarrow |0\rangle\,|q_3\rangle\,|L\rangle, & \delta(q_2,1) \rightarrow |1\rangle\,|q_3\rangle\,|L\rangle, \\
\delta(q_3,\#) \rightarrow |\#\rangle\,|q_4\rangle\,|R\rangle, & \delta(q_3,0) \rightarrow |0\rangle\,|q_2\rangle\,|L\rangle, & \delta(q_3,1) \rightarrow e^{i\frac{\pi}{4}}\,|1\rangle\,|q_2\rangle\,|L\rangle, \\
\delta(q_4,\#) \rightarrow |\#\rangle\,|q_f\rangle\,|L\rangle, & \delta(q_4,0) \rightarrow |0\rangle\,|q_f\rangle\,|L\rangle, & \delta(q_4,1) \rightarrow |1\rangle\,|q_f\rangle\,|L\rangle, \\
\delta(q_f,\#) \rightarrow |\#\rangle\,|q_0\rangle\,|R\rangle, & \delta(q_f,0) \rightarrow |0\rangle\,|q_0\rangle\,|R\rangle, & \delta(q_f,1) \rightarrow |1\rangle\,|q_0\rangle\,|R\rangle
\end{array}
$$

Similarly to the QTM previously defined for the Hadamard gate, the computation starts in the initial state q_0 with a tape head in the o^{th} cell and enters the state q_2 by moving to the right and left. Then, in the state q_2, the machine performs the phase shift operator,

$$
\begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}
$$

to the given single-qubit input state by mapping $|0\rangle \rightarrow |0\rangle$ and $|1\rangle \rightarrow e^{i\frac{\pi}{4}}\,|1\rangle$ without changing the measuring probabilities of the basis states. Since all transitions from state q_f return to initial state q_0, this can be characterised as a normal form QTM. By Theorem 2, well-formedness of this unidirectional QTM is proven.

According to the transition function defined above, we can formulate a quantum state transition diagram for the QTM that performs a phase shift of $\frac{\pi}{4}$ as shown in Figure 5.

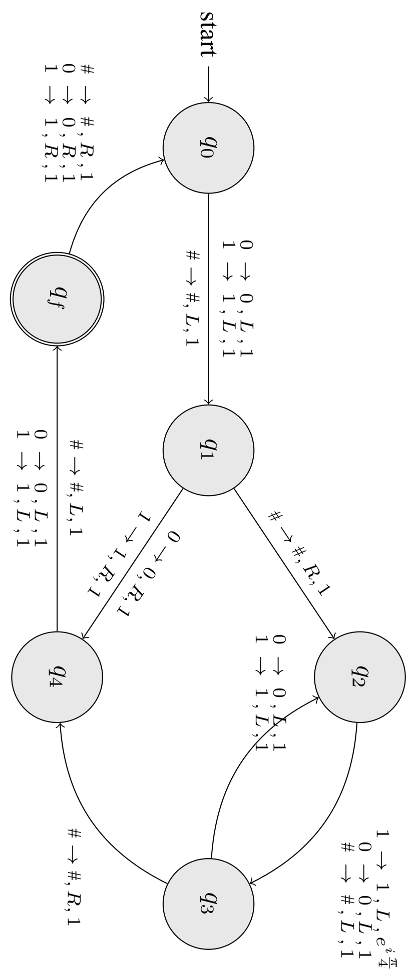

Figure 5: Quantum state transition diagram of well-defined, normal form QTM, performs T-gate transition on a single qubit.

5 Conclusion

With the recent progress of quantum computing as a computing paradigm, QTM – the the quantum counterpart of Turing machine plays a key role in modelling the effects of a quantum computer. We formally defined the QTM by incorporating the definitions of the probabilistic Turing machine, including its essential components, such as the transition function and the evolution operator. Since QTM operates under the principles of quantum mechanics, it is necessary that the designed QTM adheres to the quantum framework. For this purpose, we identified necessary conditions for a QTM to be well-formed, ensuring its consistency with the fundamentals of quantum physics. Furthermore, we recognised several variants of QTM distinguished by their unique computational behaviour. Subsequently, we examined the few previous approaches to QTM formulations. Among them, the work of [Liang and Yang 2013], which focused on a particular subclass of QTMs, called stationary rotational QTMs (SR-QTM), is notable. After investigating these SR-QTM formulations of the Hadamard and CNOT gates, we observed that their time evolution operators do not fully satisfy the unitary condition, which suggests potential limitations in the ability of such QTMs to model quantum computations in accordance with the principles of quantum physics. In the four decades following Deutsch's introduction of the Quantum Turing Machine, the literature has yet to present concrete instances of well-formed QTMs. To bridge this gap, we have chosen a set of universal fault-tolerant quantum gates, including the Hadamard, CNOT, and T gates, to construct well-formed QTMs that satisfy all necessary conditions. Additionally, we have refined the existing framework of quantum state transition diagrams (QSTDs) and proposed an improved version of QSTD that is capable of representing all varieties of QTMs without limitations.

Despite the fact that our study is primarily theoretical, we believe it has the potential for both algorithmic and computational advancements. For instance, this may provide insights into alternative implementations of non-Clifford gates, such as the T gate – known to be insubstantial and difficult to construct in a fault–tolerant setting within scalable quantum computing architectures. Moreover, just as the classical Turing machines helped in the development of computer programming, QTMs could have the potential of facilitating the growth of quantum programming. In future works, our focus lies in the development of QTM-based representations of quantum algorithms, which could lead to more efficient designs and a deeper understanding of quantum computation.

Acknowledgements

Authors would like to thank Frank Leymann and Nalin Ranasinghe for their comments on the main results in this manuscript.

References

[Bernstein and Vazirani 1997] Bernstein, E. and Vazirani, U., 1997. Quantum Complexity Theory. SIAM Journal on Computing, 26(5), pp.1411-1473.

[Bernstein and Vazirani 1993] Bernstein, Ethan, and Umesh Vazirani. "Quantum complexity theory." In Proceedings of the twenty-fifth annual ACM symposium on Theory of computing, pp. 11-20. 1993.

[Benioff 1980] Benioff, P., 1980. The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. Journal of statistical physics, 22, 563-591.

[Boykin et al. 2000] Boykin, P.O., Mor, T., Pulver, M., Roychowdhury, V. and Vatan, F., 2000. A new universal and fault-tolerant quantum basis. Information Processing Letters, 75(3), pp.101-107.

[Deutsch 1985] Deutsch, D., 1985. Quantum theory, the Church–Turing principle and the universal quantum computer. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, 400(1818), pp.97-117.

[Feynman 1982] Feynman, R.P., 1982. Simulating Physics with Computers. International Journal of Theoretical Physics, 21(6/7).

[Hook and Lee 2010] Hook IV, L.R. and Lee, S.C., 2010, March. Quantum state transition diagram: a bridge from classical computing to quantum computing. In Nanosensors, Biosensors, and Info-Tech Sensors and Systems 2010 (Vol. 7646, pp. 149-153). SPIE.

[Liang and Yang 2013] Liang, M. and Yang, L., 2013. On a class of quantum Turing machine halting deterministically. Science China Physics, Mechanics and Astronomy, 56, pp.941-946.

[Molina and Watrous 2019] Molina, A. and Watrous, J., 2019. Revisiting the simulation of quantum Turing machines by quantum circuits. Proceedings of the Royal Society A, 475(2226), p.20180767.

[Nielsen and Chuang 2010] Nielsen, M.A. and Chuang, I.L., 2010. Quantum computation and quantum information. Cambridge university press.

[Santos 1969] Santos, E.S., 1969. Probabilistic Turing machines and computability. Proceedings of the American mathematical Society, 22(3), pp.704-710.

[Yao 1993] Yao, A.C.C., 1993, November. Quantum circuit complexity. In Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science (pp. 352-361). IEEE.