Quantum Computation

Daniella E. Pombo

Abstract Based off of Dr. Guerrini and Dr. Deutsch's research, I propose a definition of a Quantum Turing Machine (QTM) comprised of quantum computation functions "mapping from superpositions to probability distributions of natural numbers" (S. Guerrini 1). In generality, QTMs simulate classical universal turing machines (UTM) as the computation of a QTM is the mapping from natural numbers to the limit of UTM's computations. This is exemplified as the final result of a QTM is the limit of parallel infinite computations of a UTM (S. Guerrini 3). Therefore, there is a relation of a QTM to UTMs as QTMs are defined by classical TM computations, functions and operations but with quantum mechanics[1] in mind. Superpositions, entanglement and quantum parallelism are similutated within a QTM through various calculus, linear algebra and statistical methods to create a machine that computes all physical systems in nature. In conclusion, a UTM is inept in its ability to model and simulate all physical systems within nature, however, a QTM, a quantum model, is equipt to model all physical systems observable and measurable within in nature.

1. Introduction

The Church-Turing Thesis (also known as the Theory of Computability) states that "'Every 'function which would naturally be regarded as computable' can be computed by the universal Turing machine'" (D. Deutsch 3). This hypothesis of what defines a computable function makes a strong assertion of computability based off of not only "Finitely realizable physical systems" (D. Deutsch 3) but continuous physical systems. A "Finitely realizable physical system" is one in which only satisfies classical physics[2]. As there are many outliers and anomalies within physics, such as within quantum physics, the word choice "Naturally regarded" leaves too much room for interpretation within a physics and mathematical model. As this assertion by the Church-Turing Thesis fails to fully address classical physics, it also fails to satisfy quantum mechanics and therefore, as quantum mechanics has now become the foundation of physics, fails to address all physical systems within nature. Therefore, Dr. David Deutsch proposes a more direct Church-Turing "Principle" stating "'Every finitely realizable physical system can be perfectly simulated by a universal model computing machine operating by finite means'"(D. Deutsch 3). By stating "Finitely realizable physical system", Dr. Deutsch refers directly to a classical physical object that can be tested, measured and observed. This interpretation of the Church Turing Thesis allows for a more direct assertion of what a physical system is as well as how it may be computable, thus leaving out the ambiguities of quantum physical systems. Therefore, Dr. Deutsch's interpretation of the Church-Turing Principle allows for a universal Turing Machine to satisfy and address only classical physics as it simulates a UTM with enhanced functions and operations. Furthermore, Dr. Deutsch's Church-Turing Principle allows for the creation and proper use of a QTM to directly address physical systems that are defined within quantum mechanics.

[1] Quantum mechanics is the microscopic study of the behavior of particles as wave-like, rather than particle-like, that behaves with an ambiguous probabilistic nature.

[2] Classical mechanics is pre modern physics in which views the behavior and movement of particles as particle-like, as thus views, describes and observes such objects at a macroscopic level.

2. Quantum Computer

A quantum computer (QC) is a computer which is defined by quantum mechanics and its QTM. A QC processes inputs and returns an output in classical bit form. As a quantum model, QCs make use of entanglement and superposition to maintain polynomial time as well as to satisfy quantum theory. However, despite the existence of capable functions that allow for classical computers to simulate QCs, QCs and classical computers are in no way similar as there is only a set of functions in which classical computers are more efficient and effective in computing rather than a QC.

Classical computers use bits to relay and maintain information between states. QCs, on the other hand, use qubits which are the smallest unit of encoded information. A qubit is a probability distribution of possible outputs and of which holds more information in comparison to its classical counterpart: for 1 qubit there are 2^n bits of information. Qubits, represented by the Bloch sphere, are linear combinations of quantum states defined by the linear superposition equation $|\varphi\rangle = a_0|0\rangle + a_1|1\rangle$ [3] and $\sum_{i=0}^{n} \left|a_i\right|^2 = 1$, where *a* denotes the probability (also referred to as amplitude in physics) of the qubit to be in the $|c\rangle$ state where $c \in \{0, 1\}$. Therefore, the linear superposition equation states that the characteristics of a qubit are the linear combination of its two orthonormal basis states[4] $|0\rangle$ [5] and $|1\rangle$ [6] which span the infinite dimensional Hilbert Space[7] such that the $\mathit{superposition} \in \mathit{hilbert\ space}\ \mathfrak{L}^2()$. Note using Born Rule[8], $\left|a_i\right|^2$ indicates taking the norm of a_i yielding its probability, and furthermore, $\langle 0|1\rangle = 0$ and $\langle 1|1\rangle = 1$ where $\langle b|c\rangle$ denotes the inner product of *b* and *c*. Due to the fundamental idea behind qubits and its quantum theory, qubits are in superpositions and are entangled. Due to the superposition principle, qubits are linear combinations of the probability distribution of a qubit to be in quantum state $|c\rangle$ and therefore reiterating that the state of the particle is simultaneously in both states of $|0\rangle$ and $|1\rangle$.

Furthermore, superpositions leads to quantum parallelism[9] which is defined as disjointed computations operating simultaneously till each reaches a final result similarly to a nondeterministic Turing machine. Thus, quantum parallelism allows for the processing and computation of large data sets to work simultaneously, thus cutting time otherwise needed.

Quantum computers are also defined by the entanglement principle which is advantageous as it allows for faster and more efficient processing time; entanglement is a physical phenomenon in which qubits are permanently connected and correlated, thus simulating system-like behavior. This is represented as the combination of qubit states which is mathematically demonstrated through a tensor product of the quantum states yielding a composite system $|a\rangle|b\rangle = |a\rangle \otimes |b\rangle = |ab\rangle$ (J. Hui) that

[3] Is also denoted as $|\varphi\rangle = a_0$[1 0]^*T* + a_1[0 1]^*T*

[4] Orthonormal basis vectors are basis vectors of which are unit vectors and orthogonal to each other, furthemore their dot product is $a \bullet b = 0$, and thus referencing the inner product space.

[5] |0⟩=[1 0]^T

[6] |1⟩= [0 1]^T

[7] The Hilbert Space is denoted as 𝔏^2(set of configurations) with unitary norm, where the hilbert space is infinite dimensional.

[8] Born rule gives the probability that the quantum state will return a certain result.

[9] Quantum parallelism is also referred to as Quantum Computational Parallelism.

simulates quantum mechanics. Therefore, by measuring a qubit that is entangled, the QC is able to deduce the qubits entangled partners’ information and behavior without having to physically and operationally measure the qubits entangled partners’ qubit value. Hence, entanglement negates the need to measured all qubits within a system thus decreasing operational and time complexities.

A QC is composed of a quantum chip and quantum logic gates which both use quantum manipulation. Furthermore both the quantum chip and quantum logic gate are similar to a classical chip and classical NAND logic gate. Both the quantum chip and quantum logic gates use quantum manipulation to physically change the qubits’ quantum states and characteristics.

Due to the nature of qubits and a QC and its QTM, qubits are only measurable when the qubits are collapsed into their bit form which is due in part to their superposition nature. Thus, qubits are not able to be directly measured which forces the QTM to implement operations that allow for measurement, readability and accessibility; the collapsing of qubits creates the need for larger operational and time complexities which creates and imposes a limitation on QTM capabilities and functionalities.

Note that the theory of a QTM relies on an utopian ideology that a QC will have a perfect fault-tolerant system and an error-correcting system in which mitigates environmental-caused errors. However, no such QC exists, yet.

3. Quantum Turing Machine

A quantum computer is defined by its QTM. A QTM is comprised of a finite controller, infinite memory, and by its the quantum computation functions which are mappings from the “Superpositions of natural numbers to [the] probability distributions of natural numbers[, where e]ach function is obtained as a limit of infinite computation of a QTM” (S. Guerrini 1) where the domain is the infinitely dimensional Hilbert space with unitary norm. Furthermore, the set of quantum computable functions is recursively enumerable, reversible, invertible and unitary[10]. In a generalized term, a QTM may be described as a UTM with parallel, disjointed, linearly independent computations that are mapped to configurations of QTMs which are superpositions expressed as a weighted sum $\Sigma d_i C_i$ of classical configurations C_i with complex coefficients d_i that relate to the Hilbert space (S. Guerrini 1). It is important to note that “Quantum configuration[s are] superposition[s] of classical configurations (S. Guerrini 4)”. Furthermore, the final result of a QTM is the limit of computations of parallel UTM computations that reach the final result which further emphasizes how QTMs and UTMs are related; thus, it is properly interpreted that many of the mechanisms, operations and ideologies of a UTM are implemented and or enhanced within a QTM.

A quantum turing machine is a 7 tuple defined as QTM M = $\boldsymbol{(} \sum, Q, Q_s, Q_t, \boldsymbol{\delta}, q_i, q_f \boldsymbol{)}$ (S. Guerrini 5) of which: $\sum$ is the input alphabet; Q are the states; Q_s is the set of source states such that $Q_s \subseteq Q$**;** Q_t is the set of target states such that $Q_t \subseteq Q$**;** $\boldsymbol{\delta}$ is the transition function**;** q_i is the start

[10] “Theorem 49. *Let* $\mathscr{V}$ *be a complex inner product vectorial space for each bounded application* $U: \mathscr{V} \to \mathscr{V}$ *there is one and only one bounded application* $U^*: \mathscr{V} \to \mathscr{V}$ *such that* $\langle x, U, y \rangle = \langle U^*x, y \rangle$*. We say that* U^* *is adjoint of* U. It is easy to show that if U is a bounded application, then U is unitary iff U is invertible and $U^* = U^{-1}$ (S. Guerrini 28). Furthermore, in quantum mechanics, unitarity of a complex matrix U is described as the product of U and its hermitian conjugate U^ † that yields the identity matrix I denoted as (U^ †)$U = I$. Furthermore, “Lemma 58. *U□ *U□=1 iff the local unitary conditions holds, that is, U□ is an isometry iff m is a QTM*” (S. Guerrini 33).

state such that $q_i \in Q_s$; q_f is the final state such that $q_f \in Q_t$; and where $\sum$, and Q are both finite. Furthermore, a QTM has a read-write head and quantum state input tape which is composed of encoded input qubit strings. A QTM is similar to a UTM with various tapes and controllers that are able to simultaneously compute infinite computations. Furthermore, it is important to note that there is a tape alphabet $\widehat{\Sigma}$ that is not indicated within the 7 tuple in Dr. Guerrini's QTM, however, note this tape alphabet is similar to our tape alphabet Γ used throughout the course.

QTMs are largely defined and limited by the fact that there are no finite computations within a QTM, implying that all computations are infinite. This is due in part as QTMs lack final and start states: there must be observable states that are designated "final" states which observe the result--"final" result--within that computation. However, despite the creation of observable states the computation will continue to execute and evolve the information within the tape before and after the observable states. As QTM functions are reversible, "Even the initial configuration must have a predecessor"(S. Guerrini 3) just as the "final" configuration will have a heir thus further complicating the question of when and how the final result should be measured. Therefore, in generality, all states, including the source and target states, within a QTM must have at minimum one incoming and one outgoing transition which includes the "initial" and "final" configurations. Note the initial configuration corresponds with the source states and the final configurations corresponds with the target states.

It was then proposed by Dr. Guerrini that the superposition of the final result of a QTM is the limit of classical UTM's configurations that yields to a final results (S. Guerrini 3), as well as, and the solution is to implement extra symbols. The purpose of the extra symbols are to manipulate the initial and final configurations in such a way as to confine them to the source or target states without the implementation of a loop as the evolution operator must be unitary. In reality, Dr. Guerrini is describing a quantum tape alphabet $\widehat{\Sigma} = \Sigma \cup \overline{\Sigma}$ where $\overline{\Sigma} = \{\overline{a} \mid a \in \Sigma\}$ (S. Guerrini 4) is the extra symbol alphabet. The implementation of the extra symbols allows the restriction of the initial and final configurations' transition functions as its implementation leads to a restriction of the transitions in and out of the source and target states. "There are no transitions out of a final state when reading a marked symbol, or which enter an initial state writing a marked symbol and no transition at all involving marked symbols" (S. Guerrini 3) entering nor exiting the non-source and non-target states. When the final state reads $a \in \Sigma$ within the tape, the controller writes $\overline{a} \in \widehat{\Sigma}$ over $a \in \Sigma$ and the read-write head proceeds to the right of the configuration while staying within the target states. Similarly, as transition functions are unitary operations, when in the initial state, if the read-write head reads to the left of the head $a \in \Sigma$, then the controller writes $\overline{a} \in \widehat{\Sigma}$ over $a \in \Sigma$, then the read-write head proceeds to the left of the configuration where it is now over the newly marked symbol; this operation simulations " rolling[ing back] to another initial configuration" (S. Guerrini 3) . Consequently, if the configuration is in the initial state, and the read-write head reads $\overline{a} \in \widehat{\Sigma}$, then the controller writes $a \in \Sigma$ over it and moves to its right, thus trapping the configuration within the initial state. Thus, through the use of extra symbols, the configuration is "trapped" within an initial or final state, and thus, the QTM is then able to measure the qubit without concern of the evolution nor contamination of information.

If there are a set of states that are non-members of the source nor target states, then it is denoted as $Q_0 = Q - (Q_s \cup Q_t)$, and hence the quantum transition function of QTM is $\delta = \delta_0 \cup \delta_s \cup \delta_t$ where $\delta_0, \delta_s, \delta_t$ have disjoint domains such that:

$$\delta_0 : ((Q_0 \cup Q_s) \times \Sigma) \rightarrow \mathfrak{L}^2((Q_0 \cup Q_t) \times \square)$$
$$\delta_s : (Q_s \times \overline{\Sigma}) \rightarrow \mathfrak{L}^2(Q_s \times \Sigma \times \square)$$
$$\delta_t : (Q_t \times \Sigma) \rightarrow \mathfrak{L}^2(Q_t \times \overline{\Sigma} \times \square)$$

Where $\square = \{L, R\}$ is the set of movement instructions L for left and R for right, and where $\mathfrak{L}^2()$ represents the Hilbert space.

The main transition function is

(a) For any $(q, a) \in (Q_0 \cup Q_s) \times \Sigma$

$$\sum_{(p,b,d) \in (Q_0 \cup Q_t) \times \Sigma \times \square} \left|\delta_0(q, a)(p, b, d)\right|^2 = 1$$

(b) For any $(q, a), (q', a') \in (Q_0 \cup Q_s) \times \Sigma$ with $(q, a) \neq (q', a')$

$$\sum_{(p,b,d) \in (Q_0 \cup Q_t) \times \Sigma \times \square} \delta_0(q', a')(p, b, d) * \delta_0(q, a)(p, b, d) = 0$$

(c) For any $(q, a, b), (q', a', b') \in (Q_0 \cup Q_s) \times \Sigma^2$

$$\sum_{(p,b,d) \in (Q_0 \cup Q_t)} \delta_0(q', a')(p, b', L) * \delta_0(q, a)(p, b, R) = 0$$

Note that $\delta_x = S_x \rightarrow \mathfrak{L}^2(\tau_x \times \square)$ for $x \in \{0, s, t\}$ and $S_x \cap S_y = \varnothing$ and $\tau_x \cap \tau_y = \varnothing$ where $x \neq y$ (S. Guerrini 5-7).

The configuration of a QTM is defined as $\langle u, q, v \rangle \in \widehat{\Sigma} * \times Q \times \widehat{\Sigma} *$ (S. Guerrini 7) which is also denoted as uqv ,where $q \in Q$ is the current state. Furthermore, *u* is the right portion of the tape, and *v* is the left portion of the tape, where *v*'s first symbol lies under the read-write head.

(a) If $q \notin Q_s \cup Q_t$, then $uv \in \Sigma *$ therefore $u, v \notin \overline{\Sigma}$ as it must satisfy the rules structured for extra symbols.

(b) If $q \in Q_s$, then $u \in \Sigma *$ and $v \in \overline{\Sigma} * \Sigma *$

(c) If $q \in Q_t$, then $v \in \Sigma *$ and $u \in \Sigma * \overline{\Sigma} *$

References

[1] B. Ömer. Components of a Quantum Computer. In *http://tph.tuwien.ac.at/~oemer/* .

[2] D. Deutsch. Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. In Proceedings of Royal Society of Longdon A 400, pages 97-117, 1985; https://people.eecs.berkeley.edu/~christos/classics/Deutsch_quantum_theory.pdf , pages 1-19, 1985.

[3] J. Hui. QC--- What are Qubits in Quantum Computing? In Medium Science https://medium.com/@jonathan_hui/qc-what-are-qubits-in-quantum-computing-cdb3cb566595 , 2018.

[4] K. Bonsor, J. Strickland. How Quantum Computers Work. In HowStuffWorks, Tech, https://computer.howstuffworks.com/quantum-computer2.htm.

[5] R. Cleve. An Introduction to Quantum Complexity Theory. In https://cds.cern.ch/record/392006/files/9906111.pdf , pages 2, 13-17, 1999.

[6] R. Coolman. What is Quantum Mechanics? In LiveScience https://www.livescience.com/33816-quantum-mechanics-explanation.html , 2014.

[7] S. Aaronson. 6.896 Quantum Complexity Theory. In MITOpenCourseware Lecture Notes, https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-845-quantum-complexity-theory-fall-2010/lecture-notes/MIT6_845F10_lec01.pdf , pages 1-3, 2008.

[8] S. Guerrini, S. Martini, A. Masini. Towards a Theory of Quantum Computability. In arXiv:1504.02817, https://arxic.org/abs/1504.2817 , pages 1-38, 2015.