

Article

Quantum Turing Machines: Computations and Measurements

Stefano Guerrini ^{1,*}, Simone Martini ^{2,3,†}  and Andrea Masini ⁴¹ LIPN, UMR 7030 CNRS, Institut Galilée, Université Sorbonne Paris Nord, 93430 Villetaneuse, France² Dipartimento di Informatica—Scienza e Ingegneria, Università di Bologna, 40126 Bologna, Italy; simone.martini@unibo.it³ Inria Sophia-Antipolis, 06902 Valbonne, France⁴ Dipartimento di Informatica, Università di Verona, 37134 Verona, Italy; andrea.masini@univr.it

* Correspondence: stefano.guerrini@univ-paris13.fr

† S. Martini: Research partially conducted while on sabbatical leave at the Collegium—Lyon Institute for Advanced Studies, 2018–2019.

Received: 5 July 2020; Accepted: 7 August 2020; Published: 11 August 2020



Abstract: Contrary to the classical case, the relation between quantum programming languages and quantum Turing Machines (QTM) has not been fully investigated. In particular, there are features of QTMs that have not been exploited, a notable example being the intrinsic infinite nature of any quantum computation. In this paper, we propose a definition of QTM, which extends and unifies the notions of Deutsch and Bernstein & Vazirani. In particular, we allow both arbitrary quantum input, and meaningful superpositions of computations, where some of them are “terminated” with an “output”, while others are not. For some infinite computations an “output” is obtained as a limit of finite portions of the computation. We propose a natural and robust observation protocol for our QTMs, which does not modify the probability of the possible outcomes of the machines. Finally, we use QTMs to define a class of quantum computable functions—any such function is a mapping from a general quantum state to a probability distribution of natural numbers. We expect that our class of functions, when restricted to classical input-output, will not be different from the set of the recursive functions.

Keywords: quantum Turing machines; computability theory; computer science; theory of programming languages

1. Introduction

It is an early recognition of computer science that programming (and programming languages) could benefit from solid semantical foundations. At the end of the 50s of the last century, both for clarity purposes and for the need of an entrance ticket into the “science club”, some prominent scientists started using mathematical techniques for the description of programming languages (e.g., Chomsky’s generative grammars, in the form of Backus-Naur Form [1]). Even more important is the proposal of general semantical models for the meaning of programs, represented at its best by McCarthy’s “mathematical theory of computation” [2], where a definition of the class of computable functions is presented as a general model for the semantics of (Algol) programs. The model is clearly idealised, in which it assumes that execution happens without physical limitations of space and time, in what will be, from that moment onwards, the standard model for programming languages, where one may assume Turing-completeness, despite each implementation being a finite state machine. The standard model builds on a long process during which Turing machines are viewed as idealised models of actual computers, and thus provide a basis for some theoretical investigation, among which the first studies on computational complexity.

Quantum computations and quantum programming languages have followed a different path. A solid and ever growing body of research is present on semantics and mathematical methods (e.g., see [3–9]), but the relations of these languages with the basic notion of (quantum) Turing machines (as introduced by Deutsch in his seminal paper on quantum computation [10]; also see [11] for a quantum-mechanical model of Turing machines) has received much less study than in the classical case. The exception is, of course, the definition of quantum Turing machines (QTMs) by Bernstein & Vazirani [12] (B&V, from now on) and its use in the establishment of a sound quantum computational complexity. However, their machines, have several strong constraints, reasonable for the purpose of a theory of computational complexity, but that do not permit to exploit all the features of the idealised model of computation that QTMs provide. In particular, meaningful computations are defined only for classical inputs (a single natural number with amplitude 1), and B&V's "well-behaved" QTMs "terminate" synchronously—either all paths in superpositions enter a final state at the same time, or all of them diverge. As a consequence, there is no chance to study—and give meaning—to infinite computations, since for B&V's "well-behaved" QTMs "non termination" corresponds to classical divergence.

Our aim is to provide a definition of QTMs with both quantum data and quantum control, hence broader than B&V's one, and to base on this notion an explicit definition of a class of functions "computed by QTMs". First of all, we allow a QTM to "start" its computation on an arbitrary quantum input—that is, a denumerable superposition of configurations, each representing a natural number. Second, because any computation of a QTM is always an infinite one, we want to give meaning also to some of these infinite computations. For such infinite behaviours, an "output" is obtained as a limit of finite portions of the computation. In order to accommodate these two aims with the unitary condition of the evolution, we have first to carefully define the very notion of QTM, in such a way that, when a computation of the machine enters into a "final state", its evolution continues to remain in that final state, without changing the output written on the tape. This kind of evolution is obtained by enriching the machines by means of a suitable counter, which plays no role during the standard evolution of the machine, but starts to be increased when a path of the computation enters into a final state (a dual treatment, by reversibility, is done for the initial state). Differently from B&V, any computation path (that is, any element of the superposition) evolves independently from the others: any path may terminate at its own time, or may diverge. Like B&V, we give local conditions ensuring that the resulting evolution respects the unitary condition.

We propose a suitable protocol for termination and observation of the result, which is robust with respect to the choice of the time instants in which the measurements for observing the results are made. This is in contrast to both B&V, where one has to know the exact termination time, and to Deutsch, where a partial measurement is made at every single step of the computation. The termination criteria for B&V and Deutsch is also one of the main obstacles for their use to define meaningful infinite computations. In all the paper, with the exception of Section 6.2 when discussing papers of other authors, we use "termination" and not "halting". Because there is no real halting for QTMs, we use a term different from the standard one used for classical TMs, to remark the difference.

Once we have a robust notion of QTMs and their observation protocol, we define the notion of "function computable by a QTM", as a mapping between superpositions of initial classical configurations to probability distributions of natural numbers, which is obtained (in general) as a limit of an infinite QTM computation. Because not all infinite computations are meaningful to the limit, the resulting class of quantum computable functions also contains non-total functions, as is usual in computability theory.

We stress that we are not proposing a kind of "super Turing computability". We expect that our class of functions, when restricted to classical input-output and computable amplitudes, will not be different from the set of the recursive functions (see the end of Section 3.5).

We believe it is interesting to have a class of machines and functions that could be seen as a simple, and standard, model of quantum programs, thus opening the door to a quantum computability theory that could be of use for the semantics of quantum programming languages.

Organization of the Paper

Section 2 introduces the central notion of Quantum Turing Machines; the technical definitions are preceded by a lengthy discussion on the motivations of the most delicate issues. We conclude the section with a theorem giving local conditions for unitarity, as in [12]. Section 3 uses QTMs to define a class of partial quantum computable functions. Section 4 deals with the problem of observables with respect to QTM computations. The protocols of Deutsch [10,12] are discussed, and a new observational protocol is proposed, proving that it agrees with the notion of computable functions of the previous section. In particular, we discuss, in detail, Deutsch's protocol, in which sense our proposal is different, and how it is related to Ozawa's termination protocol [13]. In Section 5, we propose a detailed comparison with the approach of B&V. Section 6 is a detailed discussion of related work and other approaches. Finally, we discuss in the Appendix A some additional technical issues.

2. Quantum Turing Machines

In this section, we define quantum Turing machines. Before the formal statements (Definitions 1, 3 and 4, etc.), we will discuss, in detail, some of the notions that will be introduced, to motivate and explain our technical choices. We assume the reader be familiar with classical Turing machines (otherwise, see [14]), which we assume defined with a tape infinite in both directions.

Like a classical Turing Machine (TM), a Quantum Turing Machine (QTM) has a tape (a sequence of cells) containing symbols (one in each cell) from a finite tape alphabet Σ , which includes at least the symbols 1 and \square : 1 is used to code natural numbers in unary notation, while \square is the blank symbol. We shall consider computations starting from tapes containing a sequence of $n + 1$ symbols 1 (the encoding of the natural number n); thus, in the following, for any $n \in \mathbb{N}$, we shall use \underline{n} to denote the string 1^{n+1} . By the Greek letters α, β , possibly indexed, we shall instead denote strings in Σ^* , and by $\alpha\beta$, we shall denote the concatenation of α and β . Finally, we shall use λ to denote the empty string.

2.1. Plain Configurations

The basic elements to describe the configuration of a QTM are the finite sequences of symbols on the tape (as usual, we assume that only a finite portion of the tape contains non-blank symbols), the current internal state of the machine, and the current position of the head reading a symbol in a cell of the tape. We assume that each cell of the tape has a fixed address (an integer number), corresponding to its position on the tape. That is, we assume the tape to be a function $T : \mathbb{Z} \rightarrow \Sigma$, such that, at any moment of the execution, only a finite, contiguous part of the tape is not empty, i.e., at any step of the execution, there are two constants $h \leq k$ s.t., $T(i) = \square$ for $i < h$ and $i \geq k$ (thus, when $h = k$ the tape is empty).

A canonical plain configuration of a given QTM M is a quadruple $\langle \alpha, q, \beta, i \rangle \in \Sigma^* \times Q \times \Sigma^* \times \mathbb{Z}$, s.t.:

1. $q \in Q$ is the current state, where Q is the finite set of the internal states of M .
2. $\beta \in \Sigma^*$ is the right content of the tape (w.r.t. the head position), where Σ is the tape alphabet of the machine M . If $\beta = u\beta' \neq \lambda$, the symbol $u \in \Sigma$ is the current symbol, which is the content of the current cell, while $\beta' \in \Sigma^*$ is the longest string on the tape ending with a symbol different from \square and whose first symbol (if any) is written in the cell immediately to the right of the current cell. When $\beta = \lambda$ instead, the right content of the tape, including the current cell, is empty, and the current symbol is \square .

3. α is the left content of the tape (w.r.t. the head position). That is, it is either the empty string λ , or it is the longest string on the tape starting with a symbol different from \square , and whose last symbol is written in the cell immediately to the left of the current cell.
4. i is the address of the current cell.

Given $\alpha, \beta \in \Sigma^*$, let $\alpha = a_0 a_1 \dots a_{u-1}$ and $\beta = b_0 b_1 \dots b_{v-1}$, with $u, v \geq 0$. The plain configuration $\langle \alpha, q, \beta, i \rangle$ corresponds to the tape

$$T(j) = \begin{cases} a_{j-i+u} & \text{for } i-u \leq j < i \\ b_{j-i} & \text{for } i \leq j < i+v \\ \square & \text{otherwise} \end{cases}$$

where $T(j) \neq \square$, implies that $i-u \leq j < i+v$.

In a canonical plain configuration $\langle \alpha, q, \beta, i \rangle$ the string α does not start with a \square , and β does not end with a \square . This ensures that each tape corresponds to a unique canonical plain configuration, and vice versa. However, because it will be useful to manipulate configurations that are extended with blank cells to the right (of the right content) or to the left (of the left content), we shall also consider plain configurations $\langle \alpha, q, \beta, i \rangle$ in which there is no restriction on α and β and we equate them up to the three equivalence relations that are induced by the following equations

$$\begin{aligned} \alpha &\simeq_l \square \alpha & \beta &\simeq_r \beta \square \\ \langle \alpha, q, \beta, i \rangle &\simeq \langle \alpha', q, \beta', i \rangle & \text{when } \alpha &\simeq_l \alpha' \text{ and } \beta \simeq_r \beta' \end{aligned}$$

It is readily seen that any plain configuration is equivalent to a unique canonical plain configuration, and that two configurations are equivalent if and only if they are in the same state, the address of the head is the same, and they correspond to the same tape.

2.2. Hilbert Space of Configurations

We will see in the following that a quantum configuration of a QTM cannot simply be a plain configuration, but it must be a weighted superposition of configurations: a vector of the Hilbert space $\ell^2(\mathcal{C})$, where \mathcal{C} is a set of configurations, like the plain ones defined above.

We recall that, for any denumerable set \mathcal{B} , $\ell^2(\mathcal{B})$ is the Hilbert space of square summable \mathcal{B} -indexed sequences of complex numbers

$$\left\{ \phi : \mathcal{B} \rightarrow \mathbb{C} \mid \sum_{C \in \mathcal{B}} |\phi(C)|^2 < \infty \right\}$$

equipped with an inner product $\langle . \mid . \rangle$ and the euclidean norm $\|\phi\| = \sqrt{\langle \phi \mid \phi \rangle}$, and that $\ell_1^2(\mathcal{B})$ denotes the set of vectors $\{\phi \mid \phi \in \ell^2(\mathcal{B}) \text{ \& } \|\phi\| = 1\}$.

By using Dirac notation, we shall write $|\phi\rangle$ to denote the vector of $\ell^2(\mathcal{B})$ corresponding to the function $\phi : \mathcal{B} \rightarrow \mathbb{C}$. Moreover, for every $C \in \mathcal{B}$, we shall write $|C\rangle$ to denote the vector corresponding to C , which is, the function equal to 1 on C , and equal to 0 on the other elements of \mathcal{B} . Finally, we remark that any vector $|\phi\rangle$ of $\ell^2(\mathcal{B})$ can be written as $\sum_{i \in I} d_i |C_i\rangle$, for some denumerable set of indexes I s.t. $\{C_i \mid i \in I\} \subseteq \mathcal{B}$ and $d_i \in \mathbb{C}$, for every $i \in I$. For more details on the basic notions of Hilbert spaces, see Appendix A.

2.3. Transitions of a QTM

Given a quantum configuration ϕ of a QTM M (see Section 2.7 for the formal definition), the main idea that is introduced by Deutsch [10] is that the machine evolves into another quantum configuration $|\psi\rangle = U|\phi\rangle$, where U is a unitary operator on the Hilbert space of the configurations of M . By linearity

and continuity, this also means that, if \mathcal{C} is the set of configurations on which we define the Hilbert space of quantum configuration $\ell^2(\mathcal{C})$ of M , the operator U , and then the behaviour of M , is completely determined by the value of U on the elements of \mathcal{C} . B&V [12] refines this point by assuming that, as in a classical TM, for every $C \in \mathcal{C}$, the transition from $|C\rangle$ to a new configuration $U|C\rangle$ only depends on the current state of M and on the current symbol of C , and that $U|C\rangle$ is formed of configurations obtained by replacing the current symbol u of C with a new one, and by moving the tape head to the left or the right. Therefore, if we denote by $\mathbb{D} = \{L, R\}$ the set of the possible movements on the tape (where L stands for left and R for right), and q is the current state of M , we have

$$U(|C\rangle) = \sum_{p,v,d} \delta(q,u)(p,v,d) |C_{p,v,d}\rangle,$$

where p ranges over the states of M , v ranges over its tape alphabet, $\delta(q,u)(p,v,d) \in \mathbb{C}$, and $C_{p,v,d}$ is the new configuration obtained from C by changing the current state from q to p , by replacing the current symbol u with v , and by moving the tape head in the direction d .

2.4. Initial and Final Configurations

In B&V, the transition rule described above applies to every state of the machine. However, as already remarked in the introduction, this implies a severe restriction on the machines that one can actually consider as valid ones—the reversibility of unitary operators, and the problem of how to read the output, force to ask that, in a “well-behaved” QTM, if a configuration in superposition enters the final state, then all other configurations of the superposition must also enter into the final state.

The key point is that a QTM cannot stop into some final configuration C_f , since the unitarity of U requires that $U(|C_f\rangle)$ be defined. On the other hand, we cannot assume that, after reaching some final configuration C_f , the computation loops on it neither. As this would imply that C_f could not be reached from any other configuration $D \in \mathcal{C}$: if $U(|C_f\rangle) = |C_f\rangle$, then $|C_f\rangle = U^{-1}(|C_f\rangle)$, and $U(|D\rangle)(C_f) = \langle U(|D\rangle) | C_f \rangle = \langle D | U^{-1}(|C_f\rangle) \rangle = \langle D | C_f \rangle = 0$, for $D \neq C_f$. When a final configuration C_f is reached, the machine must evolve into a different configuration that preserves the output written on the tape, without breaking the unitarity of the evolution operator U . Let us assume to have a denumerable set of indexed configurations $\langle C_f, n \rangle$ obtained by adding a counter $n \in \mathbb{N}$ to C_f , for every final configuration C_f . The configuration $\langle C_f, 0 \rangle$ plays the usual role of the plain configuration C_f and it is the only one that can be reached from a non final configuration; for every $n \in \mathbb{N}$, the configuration $\langle C_f, n+1 \rangle$ is instead the only successor of $\langle C_f, n \rangle$, which is, $U(|C_f, n\rangle) = |C_f, n+1\rangle$, where $|C_f, n\rangle$ is the base vector corresponding to $\langle C_f, n \rangle$. In other words, the counter n is initialised to 0 and plays no role while the state is not final; when a branch of the computation enters a final state, the counter is still at 0, but it is increased by 1 at each following step.

Final configurations are not the only configurations on which it would be useful to loop. In some cases, in order to assure that the operator U is unitary, we need to introduce additional target configurations that behave as sinks, from which the machine cannot get out (see Remark 1, and the example in Section 5.3, whose graph representation is given in Figure 4). We have then to consider a whole set \mathcal{Q}_t of target states, containing the final state q_f , s.t. for every plain configuration $C_t = \langle \alpha, q_t, \beta, i \rangle$, with $q_t \in \mathcal{Q}_t$, we have a set of indexed configurations $\langle C_t, n \rangle$, for $n \in \mathbb{N}$, s.t. $U(|C_t, n\rangle) = |C_t, n+1\rangle$.

Moreover, when U is unitary, its inverse U^{-1} is unitary too, and it can be applied to any initial configuration. Therefore, even if we assume that a computation always starts from some initial plain configuration $C_j = \langle \alpha, q_j, \beta, i \rangle$, such a C_j must have a predecessor $|C_{j,1}\rangle = U^{-1}(|C_j\rangle)$, and more generally a n -predecessor $|C_{j,n}\rangle = U^{-n}(|C_j\rangle)$. As already done for final configurations, we associate to every C_j a set of indexed configurations $\langle C_j, n \rangle$, with $n \in \mathbb{N}$, s.t. $U^{-1}(|C_j, n\rangle) = |C_j, n+1\rangle$, for $n \in \mathbb{N}$. For $n > 0$, we get then $U(|C_j, n\rangle) = |C_j, n-1\rangle$; while $U(|C_j, 0\rangle)$ has the usual behaviour expected from the machine on the plain initial configuration C_j . As for final state and final configurations, it is

useful to generalise the above behaviour to a set of source configurations corresponding to a set \mathcal{Q}_s of source states that contains the initial state q_i .

2.5. Pre Quantum Turing Machines

In order to formally define QTMs, we first define a notion of pre Quantum Turing Machine, or pQTM. As we shall see later (Definition 7), a QTM is a pQTM whose evolution operator is unitary. Because the behaviour on target states and on source states whose counter is greater than 0 is fixed, in order to completely describe a pQTM, it suffices to give a transition function $\delta_0 : ((\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma) \rightarrow \ell_1^2((\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D})$, which, for every configuration C with current state q and current symbol u , gives the weight $\delta_0(q, u)(p, v, d)$ of $|C_{p,v,d}\rangle$ in the superposed configuration reached from $|C\rangle$, where $C_{p,v,d}$ is obtained by replacing v for u , by moving the tape in the d direction, and by changing the current state to p .

Definition 1 (Pre Quantum Turing Machine). *Given a finite set of states \mathcal{Q} and an alphabet Σ , a pre Quantum Turing Machine (pQTM) is a tuple*

$$M = \langle \Sigma, \mathcal{Q}, \mathcal{Q}_s, \mathcal{Q}_t, \delta_0, q_i, q_f \rangle$$

where

- $\mathcal{Q}_s \subseteq \mathcal{Q}$ is the set of source states of M , and $q_i \in \mathcal{Q}_s$ is a distinguished source state named the initial state of M ;
- $\mathcal{Q}_t \subseteq \mathcal{Q}$ is the set of target states of M , and $q_f \in \mathcal{Q}_t$ is a distinguished target state named the final state of M ;
- \mathcal{Q}_s and \mathcal{Q}_t have the same cardinality and $\mathcal{Q}_s \cap \mathcal{Q}_t = \emptyset$;
- $\delta_0 : ((\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma) \rightarrow \ell_1^2((\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D})$ is the quantum transition function of M , where $\mathbb{D} = \{L, R\}$.

Remark 1 (Source and target states). *Source and target states must be introduced to complete a QTM when the transitions implementing its expected behaviour fail to assure the local conditions of Theorem 1, required to get a unitary time evolution operator. In particular, the first condition states that the norm of the vector $\delta(q, u)$ corresponding to the transition from a configuration C with current state q and current symbol u , to a superposed configuration $\sum \delta(q, u)(p, v, d) |C_{p,v,d}\rangle$ must be equal to 1. Anyhow, in some case, the norm of the vector of the weights implementing the expected behavior of the machine might give a value smaller than 1, while, in no case, it cannot exceed the value of 1. To achieve the required local condition, we can add some new transitions to some special target state acting as a sink (i.e., from which the machine cannot get out), with a procedure reminding the one used to complete a non-deterministic automata by connecting all of the omitted transitions to a sink state. The main difference is that in the QTM case more than one target state may be necessary to achieve the goal. Moreover, because of the reversibility of QTMs, a similar condition must hold for the transitions entering into a given state. Accordingly, we must also take into account the case in which we need to complete the implementation of the expected behaviour by adding new transitions from some source states. Finally, we remark that the requirement that the sets of source and target states have the same cardinality is used to prove the unitarity of the evolution operator associated to our QTMs via a reduction to the B&V QTMs (see Theorem 1). We do have another, independent, and direct proof of this result (that we omit in this paper), and that also exploits the equinumerosity of source and target states. We conjecture then that this is indeed a necessary condition.*

Remark 2 (On gödelization of QTMs). *For a definition of gödelizable QTMs, we must restrict the definition of the transition function to computable complex numbers (see also [12].)*

Definition 2 (computable numbers). *A real number x is computable if there exists a deterministic Turing machine that on input 1^n computes a binary representation of an integer $m \in \mathbb{Z}$, such that $|\frac{m}{2^n} - x| \leq \frac{1}{2^n}$. The set $\tilde{\mathbb{C}}$ of the computable complex numbers is the set of the complex numbers whose real and imaginary parts are both computable.*

Now, let $\tilde{\ell}_1^2(\mathcal{B})$ be the subset of $\ell_1^2(\mathcal{B})$ s.t. the coefficients of the vectors are bound to be in $\tilde{\mathbb{C}}$. We can say that a transition function $\tilde{\delta}_0$ is computable when

$$\tilde{\delta}_0 : ((Q \setminus Q_t) \times \Sigma) \rightarrow \tilde{\ell}_1^2((Q \setminus Q_s) \times \Sigma \times \mathbb{D}).$$

The gödelization of a QTM whose transition function is computable proceeds in the usual way. First of all, just observe that $\ell_1^2((Q \setminus Q_s) \times \Sigma \times \mathbb{D})$ is a finite dimensional space, then each complex number $x + iy \in \tilde{\mathbb{C}}$ involved in the gödelization of $\tilde{\delta}_0$ can be replaced by a pair (n_x, n_y) , where n_x and n_y are the Gödel numbers of the Turing machines computing x and y , respectively.

2.6. Configurations

We have already seen that, in order to properly deal with source and target states, we have to associate a counter to those states. For the sake of uniformity, we shall add a counter to every plain configuration. The counter will always be 0 for every non-source or non-target configuration.

Remark 3 (The counter). *The counter can be seen as an additional device (for instance, as an additional tape or as a counting register) or directly implemented by a suitable extension of the basic Turing machine (for instance, by extending the tape alphabet). None of these implementations is canonical or it has a direct influence in what will be presented in the following. Appendix B provides a more detailed discussion of the implementation of the counter.*

Definition 3 (configurations). *Let $M = (\Sigma, Q, Q_s, Q_t, \delta_0, q_0, q_f)$ be a pQTM. A configuration of M is a quintuple $\langle \alpha, q, \beta, i, n \rangle \in \Sigma^* \times Q \times \Sigma^* \times \mathbb{Z} \times \mathbb{N}$, where $\langle \alpha, q, \beta, i \rangle$ is a plain configuration, and n is a counter associated to the configuration s.t. $n = 0$, when $q \notin Q_s \cup Q_t$. A configuration of M is a source/target configuration when the corresponding state is a source/target state, and it is a final/initial configuration when the current state is final/initial. We have the following notations:*

- \mathfrak{C}_M is the set of the configurations of M .
- \mathfrak{C}_M^s and \mathfrak{C}_M^t are the sets of the source and of the target configurations of M , respectively.
- $\mathfrak{C}_M^{\text{init}}$ and $\mathfrak{C}_M^{\text{fin}}$ are the sets of the initial and final configurations of M , respectively.
- \mathfrak{C}_M^0 is the set of the configurations $\langle \alpha, q, \beta, i, 0 \rangle$ of M .

In the following, the subscript M in \mathfrak{C}_M and in the other names indexed by the machine may be dropped when clear from the context.

2.7. Quantum Configurations

The evolution of a pQTM is described by superpositions of configurations in \mathfrak{C}_M . If $\mathcal{B} \subseteq \mathfrak{C}_M$ is a set of configurations, a superposition of configurations in \mathcal{B} is a vector of the Hilbert space $\ell^2(\mathcal{B})$ (see, e.g., [15,16]). Quantum configurations of a pQTM M are the elements of $\ell_1^2(\mathfrak{C}_M)$ (namely, the unit vectors of $\ell^2(\mathfrak{C}_M)$). Because there is no bound on the size of the tape in a configuration, the set \mathfrak{C}_M is infinite and the Hilbert space of the configurations $\ell^2(\mathfrak{C}_M)$ is infinite dimensional.

Definition 4 (quantum configurations). *Let M be a pQTM. The elements of $\ell_1^2(\mathfrak{C}_M)$ are the quantum configurations (or more succinctly, q -configurations) of M .*

We shall use Dirac notation (see Appendix A) for the elements ϕ, ψ of $\ell^2(\mathfrak{C}_M)$, writing them $|\phi\rangle, |\psi\rangle$.

Definition 5 (computational basis). *For any set of configurations $\mathcal{B} \subseteq \mathfrak{C}_M$ and any $C \in \mathcal{B}$, let $|C\rangle : \mathcal{B} \rightarrow \mathbb{C}$ be the function*

$$|C\rangle(D) = \begin{cases} 1 & \text{if } C = D \\ 0 & \text{if } C \neq D. \end{cases}$$

The set $\text{CB}(\mathcal{B})$ of all such functions is a Hilbert basis for $\ell^2(\mathcal{B})$ (see, e.g., [17]). In particular, following the literature on quantum computing, $\text{CB}(\mathfrak{C}_M)$ is called the computational basis of $\ell^2(\mathfrak{C}_M)$. Each element of the computational basis is called base q -configuration.

With a little abuse of language, we shall write $|C\rangle \in |\phi\rangle$ when $\phi(C) \neq 0$. The span of $\text{CB}(\mathcal{B})$, denoted by $\text{span}(\text{CB}(\mathcal{B}))$, is the set of the finite linear combinations with complex coefficients of elements of $\text{CB}(\mathcal{B})$; $\text{span}(\mathcal{B})$ is a vector space, but not a Hilbert space. In order to get a Hilbert space from $\text{span}(\text{CB}(\mathcal{B}))$, we have to complete it, and $\ell^2(\mathcal{B})$ is indeed the unique (up to isomorphism) completion of $\text{span}(\text{CB}(\mathcal{B}))$ (see [12]). As a consequence, any bounded linear operator U on $\text{span}(\text{CB}(\mathcal{B}))$ has a unique extension on $\ell^2(\mathcal{B})$, and, when U is unitary, its extension is unitary too.

For some basic definitions, properties, and notations on Hilbert spaces with denumerable basis, see Appendix A. In particular, Appendix A.1 presents a synoptic table of the so-called Dirac notation that we shall use in the paper.

2.8. Time Evolution Operator and QTM

In order to define the evolution operator of a pQTM M , it suffices to give its behaviour on the computational basis $\text{CB}(\mathfrak{C}_M)$. In particular, we have to distinguish three cases:

1. $C \in \mathfrak{C}_M^0 \setminus \mathfrak{C}_M^t$.
Let $C_{p,v,d} \in \mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s$ be the configuration obtained by leaving the counter to 0, by replacing the symbol u in the current cell with the symbol v , by moving the head on the d direction, and by setting the machine into the new state p . In detail, if $C \simeq \langle \alpha w, q, u\beta, i, 0 \rangle$

we have

$$C_{p,v,d} \simeq \begin{cases} \langle \alpha w v, p, \beta, i+1, 0 \rangle & \text{when } d = R \\ \langle \alpha, p, w v \beta, i-1, 0 \rangle & \text{when } d = L. \end{cases}$$

and we define

$$W_{0,M}(|C\rangle) = \sum_{(p,v,d) \in (\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D}} \delta_0(q, u)(p, v, d) |C_{p,v,d}\rangle$$

where δ_0 is the quantum transition function of M , and, as already introduced at the end of Section 2.3, $C_{p,v,d}$ is the new configuration obtained from C by changing the current state from q to p , by replacing the current symbol u with v , and by moving the tape head in the direction d .

2. $C \in \mathfrak{C}_M^s \setminus \mathfrak{C}_M^0$.
Let $C_{-1} \in \mathfrak{C}_M^s$ be the source configuration obtained by decreasing by 1 the counter of C , we define

$$W_{s,M}(|C\rangle) = |C_{-1}\rangle$$

3. $C \in \mathfrak{C}_M^t$.
Let $C_{+1} \in \mathfrak{C}_M^t \setminus \mathfrak{C}_M^0$ be the target configuration obtained by increasing by 1 the counter of C , we define

$$W_{t,M}(|C\rangle) = |C_{+1}\rangle$$

We have then three linear operators

$$\begin{aligned} W_{0,M} &: \text{span}(\text{CB}(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^t)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s)) \\ W_{s,M} &: \text{span}(\text{CB}(\mathfrak{C}_M^s \setminus \mathfrak{C}_M^0)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^s)) \\ W_{t,M} &: \text{span}(\text{CB}(\mathfrak{C}_M^t)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^t \setminus \mathfrak{C}_M^0)) \end{aligned}$$

defined on three disjoint subspaces. Moreover, because even the images of these three operators are disjoint, their sum

$$W_M = W_{0,M} + W_{s,M} + W_{t,M}$$

defines an automorphism on the linear space of q-configurations

$$W_M : \text{span}(\text{CB}(\mathfrak{C})) \rightarrow \text{span}(\text{CB}(\mathfrak{C})).$$

Indeed, since W_M is bounded, it extends in a unique way to a continuous operator on the Hilbert space of q-configurations.

Definition 6 (time evolution operator). *The time evolution operator of M is the unique continuous extension*

$$U_M : \ell^2(\mathfrak{C}_M) \rightarrow \ell^2(\mathfrak{C}_M)$$

of the bounded linear operator $W_M : \text{span}(\text{CB}(\mathfrak{C})) \rightarrow \text{span}(\text{CB}(\mathfrak{C}))$.

As it is the case for the operator W_M , also the time evolution operator U_M can be decomposed into three operators

$$\begin{aligned} U_{0,M} &: \ell^2(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^t) \rightarrow \ell^2(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s) \\ U_{s,M} &: \ell^2(\mathfrak{C}_M^s \setminus \mathfrak{C}_M^0) \rightarrow \ell^2(\mathfrak{C}_M^s) \\ U_{t,M} &: \ell^2(\mathfrak{C}_M^t) \rightarrow \ell^2(\mathfrak{C}_M^t \setminus \mathfrak{C}_M^0) \end{aligned}$$

s.t.

$$U_M = U_{0,M} + U_{s,M} + U_{t,M}.$$

Definition 7 (QTM). *A pQTM is a Quantum Turing Machine (QTM) when its time evolution operator U_M is unitary.*

As usual, U_M^i (the i -th iterate of U_M) is defined as $U_M^0 = id$, and $U_M^{i+1} = U_M \circ U_M^i$.

2.9. Computations

A computation of a QTM is an iteration of its evolution operator on some q-configuration. Because the time evolution operator of a QTM is unitary, it preserves the norm of its argument and maps q-configurations into q-configurations. By the way, this also holds for the inverse U_M^{-1} of the time evolution operator.

Property 1. *Let M be a QTM. If $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M)$, then $U_M^i|\phi\rangle \in \ell_1^2(\mathfrak{C}_M)$, for every $i \in \mathbb{Z}$.*

Definition 8 (initial and final configurations). *A q-configuration $|\phi\rangle$ is initial when $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M^{\text{init}})$ and is final when $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M^{\text{fin}})$. By $|\underline{n}\rangle$, we denote the initial configuration $\langle \lambda, q_0, \underline{n}, 0, 0 \rangle \in \mathfrak{C}_M^{\text{init}}$.*

Definition 9 (computations). *Let M be a QTM and let U_M be its time evolution operator. For an initial q-configuration $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M^{\text{init}})$, the computation of M on $|\phi\rangle$ is the denumerable sequence $\{|\phi_i\rangle\}_{i \in \mathbb{N}}$ s.t.*

1. $|\phi_0\rangle = |\phi\rangle$;
2. $|\phi_i\rangle = U_M^i|\phi\rangle$.

Clearly, any computation of a QTM M is uniquely determined by its initial q-configuration. The computation of M on the initial q-configuration $|\phi\rangle$ will be denoted by $K_{|\phi\rangle}^M$.

Remark 4. *The definition of the time evolution operator ensures that the final configurations reached along a computation are stable and do not interfere with other branches of the computation in superposition, which may enter into a final configuration later. Indeed, given a configuration $|\phi\rangle = |\phi_f\rangle + |\phi_{nf}\rangle$, where $|\phi_f\rangle \in \ell^2(\mathfrak{C}_M^{\text{fin}})$ and $|\phi_{nf}\rangle$ does not contain any final configuration, let $|\psi\rangle = U^i|\phi\rangle = U^i|\phi_f\rangle + U^i|\phi_{nf}\rangle$. Any final*

configuration in $U^i|\phi_{nf}\rangle$ has a value of the counter less than i , while any final configuration $|C, k\rangle \in |\phi\rangle$ formed of a plain configuration C and a counter k is mapped into a configuration $|C, i+k\rangle \in \psi$. Moreover, $|C, k\rangle$ and $|C, i+k\rangle$ have the same coefficient in $|\phi\rangle$ and $|\psi\rangle$, respectively, since $\langle\psi|C, i+k\rangle = \langle U^i|\phi\rangle, U^i|C, k\rangle\rangle = \langle\phi|C, k\rangle$.

2.10. Local Conditions for Unitary Evolution

In analogy to the main approaches in the literature [12,18], it is possible to state a set of local condition for the quantum transition function δ_0 of pQTM in order to ensure that the time evolution operator is unitary.

Theorem 1. Let M be a pQTM with quantum transition function δ_0 . The time evolution operator $U_M : \ell^2(\mathfrak{C}_M) \rightarrow \ell^2(\mathfrak{C}_M)$ of M is unitary if and only if δ_0 satisfies the local conditions:

1. for any $(q, a) \in (\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma$

$$\sum_{(p,b,d) \in (\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D}} |\delta_0(q, a)(p, b, d)|^2 = 1$$

2. for any $(q, a), (q', a') \in (\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma$ with $(q, a) \neq (q', a')$

$$\sum_{(p,b,d) \in (\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D}} \delta_0(q', a')(p, b, d)^* \delta_0(q, a)(p, b, d) = 0$$

3. for any $(q, a, b), (q', a', b') \in (\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma^2$

$$\sum_{p \in (\mathcal{Q} \setminus \mathcal{Q}_s)} \delta_0(q', a')(p, b', L)^* \delta_0(q, a)(p, b, R) = 0$$

where $()^*$ denote the complex conjugate operator.

Proof. We show that, to any QTM M (as defined above), we can associate a B&V QTM N , which differs on the behaviour on source and target states only. The unitarity of M follows then from the unitarity of N .

Let $M = \langle \Sigma, \mathcal{Q}, \mathcal{Q}_s, \mathcal{Q}_t, \delta_0, q_i, q_f \rangle$. We take the B&V QTM N with the same states \mathcal{Q} and the same alphabet Σ , and whose transition function ρ satisfies the following conditions:

1. ρ is the same of the transition function δ_0 of M for any non-terminal state;
2. for any terminal state q_t and every current symbol u , ρ has a unique non-null transition, with weight 1, that leaves the current symbol on the tape unchanged, moves the head to the right, and goes into an initial state q_s ; and,
3. the non-null out transitions from two distinct terminal states lead to two distinct source states, defining in this way a bijection between the set of the target and source states.

In other words, ρ is defined by:

$$\begin{aligned} \rho(q, u)(p, v, d) &= \delta_0(q, u)(p, v, d) && \text{for } q \in \mathcal{Q} \setminus \mathcal{Q}_t, p \in \mathcal{Q} \setminus \mathcal{Q}_s, u, v \in \Sigma, d \in \mathbb{D} \\ \rho(q, u)(s, v, d) &= 0 && \text{for } q \in \mathcal{Q} \setminus \mathcal{Q}_t, s \in \mathcal{Q}_s, u, v \in \Sigma, d \in \mathbb{D} \end{aligned}$$

and, assuming $\mathcal{Q}_s = \{s_i\}_{0 \leq i \leq k}$ and $\mathcal{Q}_t = \{t_i\}_{0 \leq i \leq k}$

$$\begin{aligned} \rho(t_i, u)(s_i, u, R) &= 1 && \text{for } u \in \Sigma \\ \rho(t_i, u)(s_i, v, L) &= 0 && \text{for } u, v \in \Sigma \\ \rho(t_i, u)(q, v, d) &= 0 && \text{for } q \in \mathcal{Q} \setminus \mathcal{Q}_t, q \neq s_i, u, v \in \Sigma, d \in \mathbb{D} \end{aligned}$$

The above definition ensures that N is a B&V QTM whose configurations are the plain configurations of M , and s.t. its transition function ρ verifies B&V local conditions. By B&V, the evolution operator U_N of N is then a unitary operator on the Hilbert space of the plain configurations of N and, as a consequence, its adjoint U_N^* is its inverse. We can take three linear operators

$$\begin{aligned} V_{0,M} &: \text{span}(\text{CB}(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^t)) \\ V_{s,M} &: \text{span}(\text{CB}(\mathfrak{C}_M^s)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^s \setminus \mathfrak{C}_M^0)) \\ V_{t,M} &: \text{span}(\text{CB}(\mathfrak{C}_M^t \setminus \mathfrak{C}_M^0)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^t)) \end{aligned}$$

defined by

$$\begin{aligned} V_{0,M}|C\rangle &= |D\rangle && \text{with } |D\rangle = U_N^*|C\rangle \text{ when } |C\rangle \in \mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s \\ V_{s,M}|C\rangle &= |C_{+1}\rangle && \text{when } |C\rangle \in \mathfrak{C}_M^s \\ V_{t,M}|C\rangle &= |C_{-1}\rangle && \text{when } |C\rangle \in \mathfrak{C}_M^t \setminus \mathfrak{C}_M^0 \end{aligned}$$

Which are the inverses of the three corresponding linear operators $W_{0,M}$, $W_{s,M}$, and $W_{t,M}$, respectively (see Section 2.8). Thus, the bounded linear operator $V_M = V_{0,M} + V_{s,M} + V_{t,M}$ is the inverse of $W_M = W_{0,M} + W_{s,M} + W_{t,M}$, and its continuous extension (obtained by completion) is the inverse of U_M . Moreover, such an operator is also the adjoint U_M^* of U_M . Summing up, the adjoint U_M^* of U_M is its inverse and, as a consequence, U_M is unitary. \square

Remark 5. The reader may wonder why we need to have cell indexes in configurations, when they are not used in many definitions of classical TMs. In fact, cell indexes are only needed in a very special case, which is of a completely empty tape. Let us suppose, indeed, to eliminate indexes, and to feed a completely blank tape to a QTM with only two states $\{q, r\}$, with no source and final states and no tape symbols, except \square ; let the transition function δ_0 be s.t.

- $\delta_0(q, \square)(q, \square, L) = 1/2$
- $\delta_0(q, \square)(q, \square, R) = 1/2$
- $\delta_0(q, \square)(r, \square, L) = 1/2$
- $\delta_0(q, \square)(r, \square, R) = -1/2$
- $\delta_0(r, \square)(q, \square, L) = 1/2$
- $\delta_0(r, \square)(q, \square, R) = -1/2$
- $\delta_0(r, \square)(r, \square, L) = 1/2$
- $\delta_0(r, \square)(r, \square, R) = 1/2$

It is easy to check that the conditions of Theorem 1 are satisfied, but trivially the induced time evolution operator U is not an isometry, since, given the empty string λ we have $U(\lambda, q, \lambda) = U(\lambda, r, \lambda) = (\lambda, q, \lambda)$ (we omit here the counter, which always remains set to zero). The indexing we use is consistent with the one adopted by Bernstein & Vazirani, [12]—Definition 3.2, pag. 1419. Here, we just want to point out that Bernstein & Vazirani only provides a fairly light informal definition of configuration. We have preferred a more pedantic formulation, which, in our opinion, is necessary to eliminate all of the reader's doubts about the correctness of the proposed theorems.

2.11. A Comparison with Bernstein and Vazirani's QTMs: Part 1

We refer to B&V [12] for the precise definitions of the QTMs used in that paper. For the sake of readability, we informally recall the notion of what they call well formed, stationary, normal form QTMs (B&V-QTMs in the following).

A B&V-QTM $M = \langle \Sigma, \Omega, \delta, q_0, q_f \rangle$ is defined as our QTM (with one source state and one target state only) with the following differences:

1. the set of configurations coincides with all possible classical configurations, namely all the set $\Sigma^* \times \mathcal{Q} \times \Sigma^* \times \mathbb{Z}$;
2. no superposition is allowed in the initial q-configuration (it must be a classical configuration $\langle \alpha, q, \beta, i \rangle$ with amplitude 1);
3. given an initial configuration $|C\rangle$, the q-configuration $\phi_k = U_M^k |C\rangle$ is the *final q-configuration* for the computation starting in $|C\rangle$, when: (i) all of the configurations in ϕ_k are final; (ii) for all $i < k$, the q-configuration $\phi_i = U_M^i |C\rangle$ does not contain any final configuration. When this is the case, we also say that the QTM halts in k steps in ϕ_k ;
4. if a QTM halts, then the tape head is on the start cell of the initial configuration; and,
5. there is no counter and for every symbol a there is loop from the only final state q_f into the only initial state q_0 , which is, $\delta(q_f, a)(q_0, a, R) = 1$ for every $a \in \Sigma$. Therefore, because of the local unitary conditions (that must hold in the final state too), these are the only outgoing transitions from q_f , and the only incoming ones into q_0 . Accordingly, $\delta(q_f, a)(q', a', d) = 0$ if $(q', a', d) \neq (q_0, a, R)$ and $\delta(q', a')(q_0, a, d) = 0$ if $(q', a', d) \neq (q_f, a, R)$.

Theorem 2. For any B&V-QTM M there is a QTM M' s.t. for each initial configuration $|C\rangle$, if M with input $|C\rangle$ halts in k steps in a final configuration $|\phi\rangle = U_M^k |C\rangle$, then $U_{M'}^k |C\rangle = |\phi\rangle$.

Proof. The QTM M' has the same states of M , the initial state q_0 is its only source state, and the final state q_f is its only target state. Therefore, if $M = \langle \Sigma, \mathcal{Q}, \delta, q_0, q_f \rangle$, we take $M' = \langle \Sigma, \mathcal{Q}, \{q_0\}, \{q_f\}, \delta_0, q_0, q_f \rangle$, where δ_0 is the restriction of δ to $((\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma) \rightarrow \ell^2((\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D})$, which is, for each $q \neq q_f$ and $a \in \Sigma$, we have $\delta_0(q, a)(p, b, d) = \delta(q, a)(p, b, d)$, for every $(p, b, d) \in (\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D}$. Because the local unitary conditions hold for δ and, in a B&V-QTM, $\delta(q, a)(q_0, b, d) = 0$, when $q \neq q_f$, the unitary local conditions hold for δ_0 too.

By construction, if $U_M^i |C\rangle$ is not final for $0 \leq i < k$, then $|\phi_k\rangle = U_M^k |C\rangle = U_{M'}^k |C\rangle$. In particular, this holds when $|\phi_k\rangle$ is the final configuration of the B&V-QTM M . \square

We remark that the inverse cannot hold true: A B&V-QTM for a given input cannot exhibit both a non terminated computation and a terminated one (see Section 5.3 for a machine that cannot be simulated by a B&V-QTM).

3. Quantum Computable Functions

In this section, we address the problem of defining the concept of quantum computable function in an “ideal” way, without taking into account any measurement protocol. The problem of the observation protocol will be addressed in Section 4. Here, we show how each QTM naturally defines a computable function from the sphere of radius 1 in $\ell^2(\mathcal{C}_M)$ to the set of (partial) probability distributions on the set of natural numbers.

3.1. Probability Distributions

Definition 10 (Probability distributions).

1. A *partial probability distribution (PPD)* of natural numbers is a function $\mathbf{P} : \mathbb{N} \rightarrow \mathbb{R}_{[0,1]}$ such that $\sum_{n \in \mathbb{N}} \mathbf{P}(n) \leq 1$.
2. If $\sum_{n \in \mathbb{N}} \mathbf{P}(n) = 1$, \mathbf{P} is a *probability distribution (PD)*.
3. \mathbb{P} and \mathbb{P}_1 denote the sets of all the PPDs and PDs, respectively.
4. If the set $\{n : \mathbf{P}(n) \neq 0\}$ is finite, \mathbf{P} is *finite*.
5. Let $\mathbf{P}', \mathbf{P}''$ be two PPDs, we say that $\mathbf{P}' \leq \mathbf{P}''$ ($\mathbf{P}' < \mathbf{P}''$) if and only if for each $n \in \mathbb{N}$, $\mathbf{P}'(n) \leq \mathbf{P}''(n)$ ($\mathbf{P}'(n) < \mathbf{P}''(n)$).
6. Let $\mathcal{P} = \{\mathbf{P}_i\}_{i \in \mathbb{N}}$ be a denumerable sequence of PPDs; \mathcal{P} is *monotone* if and only if $\mathbf{P}_i \leq \mathbf{P}_j$, for each $i < j$.

Remark 6. In the following, we shall also use the notation $\mathbf{P}(\perp) = 1 - \sum_{n \in \mathbb{N}} \mathbf{P}(n)$. By definition, $0 \leq \mathbf{P}(\perp) \leq 1$, and a PPD is a PD if and only if $\mathbf{P}(\perp) = 0$. We also stress that \leq is a partial order of PPDs and that any PD \mathbf{P} is maximal w.r.t. to \leq .

Definition 11 (limit of a sequence of PPDs). Let $\mathcal{P} = \{\mathbf{P}_i\}_{i \in \mathbb{N}}$ be a sequence of PPDs. If, for each $n \in \mathbb{N}$, there exists $l_n = \lim_{i \rightarrow \infty} \mathbf{P}_i(n)$, we say that $\lim_{i \rightarrow \infty} \mathbf{P}_i = \mathbf{P} : \mathbb{N} \rightarrow \mathbb{R}_{[0,1]}$, with $\mathbf{P}(n) = l_n$.

3.2. Monotone Sequences of Probability Distributions

Proposition 1. Let $\mathcal{P} = \{\mathbf{P}_i\}_{i \in \mathbb{N}} \subseteq \mathbb{P}$ be a monotone sequence of PPDs.

1. $\lim_{i \rightarrow \infty} \mathbf{P}_i$ exists and it is the supremum $\sqcup \mathcal{P}$ of \mathcal{P} ;
2. $\sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) = \sqcup \{\sum_{n \in \mathbb{N}} \mathbf{P}_i(n)\}_{i \in \mathbb{N}}$;
3. $\sqcup \mathcal{P} \in \mathbb{P}$.

Proof. Because $\mathbf{P}_i(n) \leq 1$, every non-decreasing sequence $\{\mathbf{P}_i(n)\}_{i \in \mathbb{N}}$ has a supremum $\sup\{\mathbf{P}_i(n)\}_{i \in \mathbb{N}} = \lim_{i \rightarrow \infty} \mathbf{P}_i(n)$. Thus, $\lim_{i \rightarrow \infty} \mathbf{P}_i$ is defined and $\mathbf{P}_i \leq \lim_{i \rightarrow \infty} \mathbf{P}_i$, for every $i \in \mathbb{N}$. On the other hand, for any \mathbf{P}' s.t. $\mathbf{P}_i \leq \mathbf{P}'$ for $i \in \mathbb{N}$, we have $\lim_{i \rightarrow \infty} \mathbf{P}_i(n) = \sup\{\mathbf{P}_i(n)\}_{i \in \mathbb{N}} \leq \mathbf{P}'(n)$, for every $n \in \mathbb{N}$; namely, $\lim_{i \rightarrow \infty} \mathbf{P}_i \leq \mathbf{P}'$. We can then conclude (item 1 of Proposition 1) that $\sqcup \mathcal{P} = \lim_{i \rightarrow \infty} \mathbf{P}_i$.

Let us now prove item 2 of Proposition 1. First of all, since $0 \leq \mathbf{P}_i(n) \leq (\sqcup \mathcal{P})(n)$ for $i, n \in \mathbb{N}$, we have $\sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) \geq \sup \{\sum_{n \in \mathbb{N}} \mathbf{P}_i(n)\}_{i \in \mathbb{N}}$ and

$$\begin{aligned} \sup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} &\geq \sup \left\{ \sum_{n \leq k} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} \\ &= \sum_{n \leq k} \sup \{\mathbf{P}_i(n)\}_{i \in \mathbb{N}} = \sum_{n \leq k} (\sqcup \mathcal{P})(n) \end{aligned}$$

for any $k \in \mathbb{N}$. Thus,

$$\begin{aligned} \sup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} &\geq \sup \left\{ \sum_{n \leq k} (\sqcup \mathcal{P})(n) \right\}_{k \in \mathbb{N}} \\ &= \lim_{k \rightarrow \infty} \sum_{n \leq k} (\sqcup \mathcal{P})(n) = \sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) \end{aligned}$$

Summing up, we can conclude, as

$$\sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) \geq \sup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} \geq \sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n)$$

Finally, by hypothesis, $\sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \leq 1$, for any $i \in \mathbb{N}$. Therefore, $\sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) = \sup \{\sum_{n \in \mathbb{N}} \mathbf{P}_i(n)\}_{i \in \mathbb{N}} \leq 1$. Which proves (item 3 of Proposition 1) that $\sqcup \mathcal{P} \in \mathbb{P}$. \square

3.3. PPD Sequence of a Computation

The computed output of a QTM will be defined (Definition 14) as the limit of the sequence of PPDs obtained along its computations.

Definition 12 (probability and q-configurations). Given a configuration $C = \langle \alpha, q, \beta, i, n \rangle$, let $\mathbf{val}[C]$ be the number of 1s in $\alpha\beta$. For any $|\phi\rangle \in \ell^2(\mathfrak{C})$, let us define $\mathbf{P}_{|\phi\rangle} : \mathbb{N} \rightarrow \mathbb{R}_{\geq 0}$ s.t.

$$\mathbf{P}_{|\phi\rangle}(n) = \sum_{C \in \mathfrak{C}^{fin}, \mathbf{val}[C]=n} |e_C|^2$$

when $|\phi\rangle = \sum_{C \in \mathcal{C}} e_C |C\rangle$.

Proposition 2. *If $|\phi\rangle$ is a q -configuration, $\mathbf{P}_{|\phi\rangle}$ is a PPD. Moreover, it is a PD if and only if $|\phi\rangle$ is final.*

Proof. Let $|\phi\rangle = |\phi_0\rangle + |\phi_f\rangle$ with $|\phi_0\rangle \in \ell^2(\mathcal{C}_M \setminus \mathcal{C}_M^{fin})$, and $|\phi_f\rangle \in \ell^2(\mathcal{C}_M^{fin})$. By the definition of $\mathbf{P}_{|\phi\rangle}$, it is readily seen that $\mathbf{P}_{|\phi\rangle} = \mathbf{P}_{|\phi_f\rangle}$ and that $\sum_{n \in \mathbb{N}} \mathbf{P}_{|\phi\rangle}(n) = \sum_{n \in \mathbb{N}} \mathbf{P}_{|\phi_f\rangle}(n) = \|\phi_f\| \leq 1$. Therefore, $\mathbf{P}_{|\phi\rangle}$ is a PPD. Moreover, since $\|\phi_f\| = 1$ if and only if $\|\phi_0\| = 0$. We see that $\mathbf{P}_{|\phi\rangle}$ is a PD if and only if $|\phi_0\rangle = 0$, that is, if and only if $|\phi\rangle = |\phi_f\rangle$. \square

Definition 13. *For any q -configuration $|\phi\rangle$, we shall say that $\mathbf{P}_{|\phi\rangle}$ is the PPD associated to $|\phi\rangle$, and we shall denote by $\mathbf{P}_{K_{|\phi\rangle}^M}$ the sequence of PPDs $\{\mathbf{P}_{|\phi_i\rangle}\}_{i \in \mathbb{N}}$ associated to the computation $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$.*

The PPD sequence of any QTM computation is monotone. In the simple proof of this key property (Theorem 3) we see at work all of the constraints on the time evolution of a QTM M .

1. That when in a final (target) configuration, the machine can only increment the counter; as a consequence, the **val** of final (target) configurations does not change.
2. That when entering for the first time into a final (target) state, the value of the counter is initialised to 0.
3. That when in a final (target) configuration $|\phi, n\rangle$, the counter gives the number of steps n since M is looping into the plain configuration ϕ .

We stress that the last two properties defuse quantum interference between final configurations reached in a different number of steps.

Theorem 3 (monotonicity of computations). *For any computation $K_{|\phi\rangle}^M$ of a QTM M , the sequence of PPDs $\mathbf{P}_{K_{|\phi\rangle}^M}$ is monotone.*

Proof. Let us prove that $\mathbf{P}_{|\phi\rangle} \leq \mathbf{P}_{U|\phi\rangle}$, for every $|\phi\rangle \in \ell^2(\mathcal{C}_M)$. Let $|\phi\rangle = |\phi_0\rangle + \sum_{k \in \mathbb{N}} |\phi_{f,k}\rangle$, with $|\phi_0\rangle \in \ell^2(\mathcal{C}_M \setminus \mathcal{C}_M^{fin})$, and $|\phi_{f,k}\rangle \in \ell^2(\mathcal{C}_M^{fin} \cap \mathcal{C}_M^k)$, where \mathcal{C}_M^k is the set of the configurations of M whose counter is equal to $k \in \mathbb{N}$. For every $n \in \mathbb{N}$, we see that $\mathbf{P}_{|\phi\rangle}(n) = \sum_{k \in \mathbb{N}} \mathbf{P}_{|\phi_{f,k}\rangle}(n)$. Let $|\psi\rangle = |\psi_0\rangle + \sum_{k \in \mathbb{N}} |\psi_{f,k}\rangle = U|\phi\rangle$. By the definition of U , we see that $U|\phi_0\rangle = |\psi_0\rangle + |\psi_{f,0}\rangle$ and $U|\phi_{f,k}\rangle = |\psi_{f,k+1}\rangle$. Therefore, $\mathbf{P}_{|\psi\rangle}(n) = \sum_{k \in \mathbb{N}} \mathbf{P}_{|\psi_{f,k}\rangle}(n) = \mathbf{P}_{|\psi_{f,0}\rangle}(n) + \sum_{k \in \mathbb{N}} \mathbf{P}_{|\phi_{f,k}\rangle}(n) = \mathbf{P}_{|\psi_{f,0}\rangle}(n) + \mathbf{P}_{|\phi\rangle}(n) \geq \mathbf{P}_{|\phi\rangle}(n)$. \square

3.4. Computed Output

We can now come back to the definition of the computed output of a QTM computation. The easy case is when a computation reaches a final q -configuration $|\psi\rangle \in \mathcal{C}_M^{fin}$ (meaning that all of the classical computations in superposition are “terminated”)—in this case the computed output is the PD $\mathbf{P}_{|\psi\rangle}$. The QTM keeps computing and transforming $|\psi\rangle$ into other configurations, but all of these configurations have the same PD. However, we want to give meaning also to infinite computations, which never reach a final q -configuration, yet producing some final configurations in the superpositions. For this purpose, we define the computed output as the limit of the PPDs yielded by the computation.

Definition 14 (computed output of a QTM). *Let $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ be the computation of the QTM M on the initial q -configuration $|\phi\rangle$. The computed output of M on the initial q -configuration $|\phi\rangle$ is the PPD $\mathbf{P} = \lim_{i \rightarrow \infty} \mathbf{P}_{|\phi_i\rangle}$, which we shall also denote by $\lim K_{|\phi\rangle}^M$, or by the notation $M_{|\phi\rangle} \rightarrow \mathbf{P}$.*

Let us remark that, by Proposition 1 and Theorem 3, the limit in the above definition is well-defined for any computation. Therefore, a QTM has a computed output for any initial q -configuration $|\phi\rangle$.

Definition 15 (finitary computations). Given a QTM M , a q -configuration $|\phi\rangle$ is finite if it is an element of $\text{span}(\text{CB}(\mathcal{C}_M))$. A computation $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ is finitary with computed output \mathbf{P} if there exists a k s.t. $|\phi_k\rangle$ is final and $\mathbf{P}_{|\phi_k\rangle} = \mathbf{P}$.

Proposition 3. Let $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ be a finitary computation with computed output \mathbf{P} , which is, $M_{|\phi\rangle} \rightarrow \mathbf{P}$.

1. There exists a k , such that, for each $j \geq k$, $|\phi_j\rangle$ is final and $\mathbf{P}_{|\phi_j\rangle} = \mathbf{P}$.
2. \mathbf{P} is a PD.

Proof. By definition, there is a k s.t. $|\phi_k\rangle$ is final. Let $\mathbf{P}_{|\phi_k\rangle} = \mathbf{P}$. By Proposition 2, \mathbf{P} is a PD. By monotonicity, $\mathbf{P} \leq \mathbf{P}_{|\phi_j\rangle}$, for every $j \geq k$. Thus, because any PD is maximal for \leq (see Remark 6), $\mathbf{P} = \mathbf{P}_{|\phi_j\rangle}$, for every $j \geq k$. \square

Given a computation $K_{|\phi\rangle}^M$, we can then distinguish the following cases:

1. $K_{|\phi\rangle}^M$ is finitary. In this case, $M_{|\phi\rangle} \rightarrow \mathbf{P} \in \mathbb{P}_1$; the output of the computation is then a PD and is determined after a finite number of steps;
2. $K_{|\phi\rangle}^M$ is not finitary, but $M_{|\phi\rangle} \rightarrow \mathbf{P} \in \mathbb{P}_1$. The output is a PD and is determined as a limit; and,
3. $K_{|\phi\rangle}^M$ is not finitary, and $M_{|\phi\rangle} \rightarrow \mathbf{P} \in \mathbb{P} \setminus \mathbb{P}_1$ (the sum of the probabilities of observing natural numbers is $p < 1$). Not only the result is determined as a limit, but we cannot extract a PD from the output.

The first two cases above give rise to what Definition 16 calls a q -total function. However, observe that, for an external observer, cases (2) and (3) are in general indistinguishable, since, at any finite stage of the computation, we may only observe a finite part of the computed output.

For some examples of QTMs and their computed output, see Section 5.

3.5. Quantum Partial Computable Functions

We want our quantum computable functions to be defined over a natural extension of the natural numbers. When using a QTM for computing a function, we stipulate that initial q -configurations are superpositions of initial classical (plain) configurations $|C_n\rangle$, where the tape of the configuration C_n encodes the number n . A natural choice for this encoding corresponds to take $C_n = |\underline{n}\rangle = |\langle \lambda, q_0, \underline{n}, 0, 0 \rangle\rangle$, where \underline{n} denotes the string 1^{n+1} . Such q -configurations are naturally isomorphic to the space $\ell_1^2(\mathbb{N}) = \{\phi : \mathbb{N} \rightarrow \mathbb{C} \mid \sum_{n \in \mathbb{N}} |\phi(n)|^2 = 1\}$ of square summable, denumerable sequences with unitary norm, under the bijective mapping $\nu(\sum d_k n_k) = \sum d_k |n_k\rangle$. Observe that, for output, we simply count the number of (non contiguous) 1s on the tape, according to Definition 12.

Definition 16 (partial quantum computable functions).

1. A function $f : \ell_1^2 \rightarrow \mathbb{P}$ is partial quantum computable (q -computable) if there exists a QTM M s.t. $f(\mathbf{x}) = \mathbf{P}$ if and only if $M_{\nu(\mathbf{x})} \rightarrow \mathbf{P}$.
2. A q -partial computable function f is quantum total (q -total) if for each \mathbf{x} , $f(\mathbf{x}) \in \mathbb{P}_1$.

It is out of the scope of the paper to investigate in details how this class exactly relates to other classes of computable functions when the input of QTMs is restricted to classical inputs. However, observe, that when the output is also required to be classical, and all amplitudes are computable, we expect to obtain the classical (partial) recursive functions. Indeed, any classical deterministic TM may be simulated by a reversible TM [19]; moreover, B&V show that reversible TMs are a special case of QTMs (Section 4 of [12]; see also our Section 5.2). On the other direction, when we have classical inputs and computable amplitudes, it is easy to simulate a QTM via a classical TM.

4. Observables

While the evolution of a closed quantum system (e.g., a QTM) is reversible and deterministic once its evolution operator is known, a (global) measurement of a q-configuration is an irreversible process, which causes the collapse of the quantum state to a new state. Technically, a measurement corresponds to a projection on a subspace of the Hilbert space of quantum states. For the sake of simplicity, in the case of QTMs, let us restrict to measurements observing if a configuration belongs to the subspace described by some set of base configurations \mathcal{B} . The effect of such a measurement is summarised by the following:

Measurement

Given a set of configurations $\mathcal{B} \subseteq \mathcal{C}$, a measurement observing if a quantum configuration $|\phi\rangle = \sum_{C \in \mathcal{C}} e_C |C\rangle$ belongs to the subspace generated by $\text{CB}(\mathcal{B})$ gives a positive answer with a probability $p = \sum_{C \in \mathcal{B}} |e_C|^2$, equal to the square of the norm of the projection of $|\phi\rangle$ onto $\ell^2(\mathcal{B})$, causing at the same time a collapse of the configuration into the normalised projection $\sum_{C \in \mathcal{B}} p^{-\frac{1}{2}} e_C |C\rangle$; dually, with probability $1 - p = \sum_{C \notin \mathcal{B}} |e_C|^2$, it gives a negative answer and a collapse onto the subspace $\ell^2(\mathcal{C} \setminus \mathcal{B})$ orthonormal to $\ell^2(\mathcal{B})$, which is, into the normalised configuration $\sum_{C \notin \mathcal{B}} (1 - p)^{-\frac{1}{2}} e_C |C\rangle$.

This kind of measurement is a special case of the so called projective measurement that is a consequence of the postulates of quantum mechanics. For a detailed discussion, see e.g., [8,20].

Because of the irreversible modification produced by any measurement on the current configuration and, therefore, on the rest of the computation, we must deal with the problem of how to read the result of a computation. In other words, we need to establish some protocol to observe when a QTM has eventually reached a final configuration, and to read the corresponding result.

4.1. The Approach of Bernstein and Vazirani

We already discussed how B&V's "sensible" QTMs are machines where all of the computations in superposition are in some sense terminating, and reach the final state at the same time (are "stationary", in their terminology). More precisely, Definition 3.11 of B&V reads: "A final configuration of a QTM is any configuration in [final] state. If, when QTM M is run with input x , at time T the superposition contains only final configurations, and at any time less than T the superposition contains no final configuration, then M halts with running time T on input x ."

This is a good definition for a theory of computational complexity (where the problems are classical, and the inputs of QTMs are always classical) but it is of limited use in the semantics of quantum programming languages. Indeed, inputs of a B&V-QTM must be classical—we cannot extend by linearity a B&V-QTM on inputs in ℓ_1^2 , since there is no guarantee whatsoever that on different inputs the same QTM halts with the same running time.

4.2. The Approach of Deutsch

Deutsch [10] assumes that QTMs are enriched with a termination bit T . At the beginning of a computation, T is set to 0, then, the machine sets this termination bit to 1 when it enters into a final configuration. If we write $|T = i\rangle$ for the function that returns 1 when the termination bit is set to i , and 0 otherwise, a generic q-configuration of a Deutsch's QTM can be written as

$$|\phi\rangle = |T = 0\rangle \otimes \sum_{C \in \mathcal{C}^{fin}} e_C |C\rangle + |T = 1\rangle \otimes \sum_{D \in \mathcal{C}^{fin}} d_D |D\rangle$$

The observer periodically measures T in a non destructive way (that is, without modifying the rest of the state of the machine).

1. If the result of the measurement of T gives the value 0, $|\phi\rangle$ collapses (with a probability equal to $\sum_{C \notin \mathcal{C}^{fin}} |e_C|^2$) to the q-configuration

$$|\psi'\rangle = \frac{|T=0\rangle \otimes \sum_{C \notin \mathcal{C}^{fin}} e_C |C\rangle}{\sqrt{\sum_{C \notin \mathcal{C}^{fin}} |e_C|^2}}$$

and the computation continues with $|\psi'\rangle$.

2. If the result of the measurement of T gives the value 1, $|\phi\rangle$ collapses (with probability $\sum_{D \in \mathcal{C}^{fin}} |d_D|^2$) to

$$|\psi''\rangle = \frac{|T=1\rangle \otimes \sum_{D \in \mathcal{C}^{fin}} d_D |D\rangle}{\sqrt{\sum_{D \in \mathcal{C}^{fin}} |d_D|^2}}$$

and, immediately after the collapse, the observer makes a further measurement of the component $\frac{\sum_{D \in \mathcal{C}^{fin}} d_D |D\rangle}{\sqrt{\sum_{D \in \mathcal{C}^{fin}} |d_D|^2}}$ in order to read-back a final configuration.

Note that Deutsch's protocol (in an irreversible way) spoils at each step the superposition of configurations. The main point of Deutsch's approach is that a measurement must be performed immediately after some computation enters into a final state. In fact, since, at the following step, the evolution might lead the machine to exit the final state modifying the content of the tape, we would not be able to measure this output at all. In other words, either the termination bit acts as a trigger that forces a measurement each time it is set, or we perform a measurement after each step of the computation.

4.3. Problems with Deutsch Measurement

Because a QTM must be physically feasible, at least "in principle", let us analyse, in detail, whether the termination protocol proposed by Deutsch [10] is realizable.

As any quantum system, a QTM evolves with respect to a continuous time. This means that the time evolution operator U of a machine M is the solution of the Schrödinger equation with a given Hamiltonian H and depends on the time [21]. In particular, when H is constant, the solution of the equation is

$$U_t = e^{-(i/\hbar)Ht} \quad \text{for } t \in \mathbb{R}^+$$

Now, let Δ be the duration of a single step of a Deutsch QTM. This means that after n steps of the QTM, we are at time $t = n\Delta$, and that for the corresponding configurations $|\psi_n\rangle$ of the QTM, it holds the following property

$$\begin{aligned} |\psi_{n+1}\rangle &= U_{(n+1)\Delta} |\psi_0\rangle \\ &= e^{-(i/\hbar)H(n+1)\Delta} |\psi_0\rangle = e^{-(i/\hbar)H\Delta} e^{-(i/\hbar)Hn\Delta} |\psi_0\rangle = U_\Delta U_{n\Delta} |\psi_0\rangle \\ &= U_\Delta |\psi_n\rangle \end{aligned}$$

In other words, at each step n , the next configuration $|\psi_{n+1}\rangle$ is obtained by applying the evolution operator U_Δ to the current configuration $|\psi_n\rangle$.

Now, let us suppose that, as a consequence of the measurement of the halt qubit T , after n steps, and, therefore, at time $t_n = n\Delta$, the system collapses to the quantum configuration

$$|\psi_n\rangle = \frac{|T=1\rangle \otimes \sum_{D \in \mathcal{C}^{fin}} d_D |D\rangle}{\sqrt{\sum_{D \in \mathcal{C}^{fin}} |d_D|^2}}$$

According to Deutsch's approach, one has to perform a second measurement to read the actual value of the output on the tape. However, since we are in a continuous time setting, what is the time of

this second measurement? Since it can be done only after the measure of the halt qubit T , this must take place at a time $t_{n+\epsilon} = (n + \epsilon)\Delta$, with $\epsilon > 0$. We can distinguish two cases (analogous reasonings apply for $\epsilon > 1$):

1. $0 < \epsilon < 1$. In this case, the tape is measured at an intermediate time between t_n and t_{n+1} , and the machine is in an intermediate configuration $|\psi_{n+\epsilon}\rangle$ between $|\psi_n\rangle$ and $|\psi_{n+1}\rangle$. Because the QTM is defined in terms of the unitary operator U_Δ , its configurations are well-defined only at time t_n , for $n \in \mathbb{N}$. The quantum configuration $|\psi_{n+\epsilon}\rangle$ is therefore unknown and depends on how the unitary operator U_Δ is physically realised. In any case, in general it differs from $|\psi_n\rangle$.
2. $\epsilon = 1$. In this case the second measurement takes place at time t_{n+1} , on the configuration $|\psi_{n+1}\rangle = U_\Delta|\psi_n\rangle$. That implies $|\psi_n\rangle \neq |\psi_{n+1}\rangle$, since $|\psi_n\rangle = |\psi_{n+1}\rangle$ may hold only in the trivial case the unitary operator U_Δ is the identity and the $|\psi_n\rangle = |\psi_0\rangle$, for every n .

As a consequence, without additional restrictions on its evolution operator, Deutsch's approach does not allow for extracting the result of the computation of a QTM.

4.4. Fixing the Measurement Protocol

Ozawa was the first to notice the above problem. In [13], he gave a revised formulation of the measurement protocol:

Once the halt qubit is set to the state $|T = 1\rangle$, the quantum Turing machine no longer changes the halt qubit or the tape string.

However, since such a constraint would be too strong (as already remarked, if $|\psi_{n+\Delta}\rangle = |\psi_n\rangle$, then U_Δ must be the identity), Ozawa adds the footnote 11:

If the relevant outcome of the computation is designed to be written by the program in a restricted part of the tape, this condition can be weakened so that the quantum Turing machine may change the part of the tape string except that part of the tape.

In other words, instead of requiring that all tape stays unchanged after reaching a final configuration, only the interesting part of the output must become stable after the halt qubit is set to 1.

Unfortunately, Linden and Popescu maybe skipped that note, and in their unpublished paper [22] claimed that even Ozawa's was unsatisfactory. Accordingly, a few years later, Ozawa replied harshly in [23]:

Recently, Linden and Popescu [22] claimed that the halt scheme given in [13] is not consistent with unitarity of the evolution operator. However, their argument applies only to the special case in which the whole tape is required not to change after the halt. As suggested in footnote 11 of [13], the conclusion in [13] can be obtained from the weaker condition for the general case where the tape is allowed to change except for the data slot. Linden and Popescu [22] disregarded this case and hence their conclusion is not generally true.

Although Ozawa was right, he did not explain how to stabilise what he called the data slot. The approach that we shall present in the following is then a sort of implementation of Ozawa's protocol, for which we also prove in details the irrelevance of the number of executed measurements and of the measurement time.

4.5. Our Approach

The measurement of the output computed by our QTMs can be performed by following a variant of Deutsch's approach. Because of the particular structure of the transition function of our QTMs, we shall see that we do not need any additional termination bit, that a measurement can be performed at any moment of the computation, and that, indeed, we can perform several measurements at distinct

points of the computation without altering the result (in terms of the probabilistic distribution of the observed output).

Given a q-configuration $|\phi\rangle = |\phi_f\rangle + |\phi_{nf}\rangle$, where $|\phi_f\rangle \in \ell^2(\mathfrak{C}^{fin})$ and $|\phi_{nf}\rangle \in \ell^2(\mathfrak{C} \setminus \mathfrak{C}^{fin})$, our output measurement tries to get an output value from $|\phi\rangle$ by the following procedure:

1. first of all, we observe the final states of $|\phi\rangle$, forcing the q-configuration to collapse either into the final q-configuration $|\phi_f\rangle / \|\phi_f\rangle\|$, or into the q-configuration $|\phi_{nf}\rangle / \|\phi_{nf}\rangle\|$, which does not contain any final configuration;
2. then, if the q-configuration collapses into $|\phi_f\rangle / \|\phi_f\rangle\|$, we observe one of these configurations, say $|C\rangle$, which gives us the observed output $\mathbf{val}[C] = n$, forcing the q-configuration to collapse into the final base q-configuration $(e_C / |e_C|)|C\rangle$;
3. otherwise, we leave unchanged the q-configuration $|\phi_{nf}\rangle / \|\phi_{nf}\rangle\|$ obtained after the first observation, and we say that we have observed the special value \perp .

Summing up, an output measurement of $|\phi\rangle$ may lead to observe an output value $n \in \mathbb{N}$ associated to a collapse into a base final configuration $|C\rangle \in |\phi\rangle$ s.t. $\mathbf{val}[C] = n$ or to observe the special value \perp associated to a collapse into a q-configuration, which does not contain any final configuration.

Definition 17 (output observation). *An output observation with collapsed q-configuration $|\psi\rangle$ and observed output $x \in \mathbb{N} \cup \{\perp\}$ is the result of an output measurement of the q-configuration $|\phi\rangle = \sum_{C \in \mathfrak{C}} e_C |C\rangle$. Therefore, it is a triple $|\phi\rangle \downarrow_x |\psi\rangle$ s.t.*

1. either $x = n \in \mathbb{N}$, and

$$|\psi\rangle = \frac{e_C}{|e_C|} |C\rangle \quad \text{with} \quad C \in \mathfrak{C}^{fin} \text{ and } \mathbf{val}[C] = n \text{ and } e_C \neq 0$$

2. or $x = \perp$, and

$$|\psi\rangle = \frac{|\phi_{nf}\rangle}{\|\phi_{nf}\rangle\|} \quad \text{where} \quad |\phi_{nf}\rangle = \sum_{C \notin \mathfrak{C}^{fin}} e_C |C\rangle \text{ and } |\phi_{nf}\rangle \neq 0$$

The probability of an output observation is defined by

$$\Pr\{|\phi\rangle \downarrow_x |\psi\rangle\} = \begin{cases} |e_C|^2 & \text{if } x = n \in \mathbb{N} \\ \|\phi_{nf}\rangle\|^2 & \text{if } x = \perp \end{cases}$$

Remark 7. For $e \in \mathbb{C}$, $e|C\rangle \downarrow_x |\phi\rangle$, with $C \in \mathfrak{C}^{fin}$ and $\mathbf{val}[C] = n$. Since $e|C\rangle$ is a q-configuration, $|e| = 1$. By the definition of output observation, $x = n$ and $|\phi\rangle = (e/|e|)U|C\rangle = e|D\rangle$, with $|D\rangle = U|C\rangle \in \mathbf{CB}(\mathfrak{C}^{fin})$ and $\mathbf{val}[D] = n$. Moreover, $\Pr\{e|C\rangle \downarrow_x |\phi\rangle\} = |e|^2 = 1$.

Remark 8. For every pair $|\phi\rangle \downarrow_{x_1} |\psi_1\rangle$ and $|\phi\rangle \downarrow_{x_2} |\psi_2\rangle$ of distinct output observations, ψ_1 and ψ_2 are in the orthonormal subspaces generated by the two disjoint sets $\mathcal{B}_1, \mathcal{B}_2 \subseteq \mathfrak{C}$, where $\mathcal{B}_i = \{C \in \mathfrak{C} \mid |C\rangle \in |\psi_i\rangle\}$.

Definition 18 (observed run). Let M be a QTM and U_M be its time evolution operator. For any monotone increasing function $\tau : \mathbb{N} \rightarrow \mathbb{N}$ (that is, $\tau(i) < \tau(j)$ for $i < j$):

1. a τ -observed run of M on the initial q-configuration $|\phi\rangle$ is a sequence $\{|\phi_i\rangle\}_{i \in \mathbb{N}}$ s.t.:

- (a) $|\phi_0\rangle = |\phi\rangle$;
- (b) $U_M|\phi_h\rangle \downarrow_{x_i} |\phi_{h+1}\rangle$, when $h = \tau(i)$ for some $i \in \mathbb{N}$;
- (c) $|\phi_{h+1}\rangle = U_M|\phi_h\rangle$ otherwise.

2. A finite τ -observed run of length k is any finite prefix of length $k + 1$ of some τ -observed run. Notation: if $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$, then $R[k] = \{|\phi_i\rangle\}_{i \leq k}$.

In particular, let id be the identity function on \mathbb{N} . In the corresponding id -observed run, a measurement of the tape is performed after each step of the machine.

Remark 9. We stress that, given a τ -observed run $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$:

1. either it never obtains a value $n \in \mathbb{N}$ as the result of an output observation, and then it never reaches a final configuration; and,
2. or it eventually obtains such a value collapsing the q -configuration into a base final configuration $e|C\rangle$ s.t. $|e| = 1$ and $\mathbf{val}[C] = n$, and from that point onward all the configurations of the run are base final configurations $e|C_j\rangle = eU^j|C\rangle$ s.t. $\mathbf{val}[C_j] = n$, and all the following observed outputs are equal to n (see Remark 7).

Definition 19. Let $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ be a τ -observed run.

1. The sequence $\{x_i\}_{i \in \mathbb{N}}$ s.t. $|\phi_h\rangle \downarrow_{x_i} |\phi_{h+1}\rangle$, with $h = \tau(i)$, is the output sequence of the τ -observed run R .
2. The observed output of R is the value $x \in \mathbb{N} \cup \{\perp\}$ (notation: $R \downarrow_x$) defined by:
 - (a) $x = n \in \mathbb{N}$, if $x_i = n$ for some $i \in \mathbb{N}$;
 - (b) $x = \perp$ otherwise.
3. For any k , the output sequence of the finite τ -observed run $R[\tau(k) + 1]$ is the finite sequence $\{x_i\}_{i \leq k}$ and x_k is its observed output.

Definition 20 (probability of a run). Let $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ be a τ -observed run.

1. For $k \in \mathbb{N}$, the probability of the finite τ -observed run $R[k]$ is inductively defined by
 - (a) $\Pr\{R[0]\} = 1$;
 - (b) $\Pr\{R[k + 1]\} = \begin{cases} \Pr\{R[k]\} \Pr\{|\phi_k\rangle \downarrow_{x_i} |\phi_{k+1}\rangle\} & \text{when } k = \tau(i) \text{ for some } i \in \mathbb{N} \\ \Pr\{R[k]\} & \text{otherwise} \end{cases}$
2. $\Pr\{R\} = \lim_{k \rightarrow \infty} \Pr\{R[k]\}$.

We remark that $\Pr\{R\}$ is well-defined, since $1 \geq \Pr\{R[i]\} \geq \Pr\{R[j]\} > 0$, for every $i \leq j$. Therefore,

$$1 \geq \Pr\{R\} = \lim_{k \rightarrow \infty} \Pr\{R[k]\} = \inf\{\Pr\{R[k]\}\}_{k \in \mathbb{N}} \geq 0.$$

Remark 10. Let $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ be a τ -observed run s.t. $R \downarrow_n$, for some $n \in \mathbb{N}$. As observed in Remark 9, for some k , we have $R[\tau(k)] \downarrow_n$ and $R[\tau(j)] \downarrow_\perp$, for $j < k$; moreover, for $i > \tau(k)$, $|\phi_i\rangle = e|C_i\rangle$ with $|e| = 1$, $C_i \in \mathfrak{C}^{fin}$, and $\mathbf{val}[C_i] = n$. As a consequence, $\Pr\{R[\tau(k) + 1]\} = \Pr\{R[i]\}$, for any $i > \tau(k)$ (since, by Remark 7, $\Pr\{|\phi_{\tau(i)}\rangle \downarrow_n |\phi_{\tau(i)+1}\rangle\} = 1$) and $\Pr\{R\} = \lim_{i \rightarrow \infty} \Pr\{R[i]\} = \Pr\{R[\tau(k) + 1]\}$.

Definition 21 (observed computation). The τ -observed computation of a QTM M on the initial q -configuration $|\phi\rangle$, is the set $\mathcal{K}_{|\phi\rangle, \tau}^M$ of the τ -observed runs of M on $|\phi\rangle$ with the measure $\Pr : \mathcal{P}(\mathcal{K}_{|\phi\rangle, \tau}^M) \rightarrow \mathbb{C}$ (where $\mathcal{P}(\mathcal{K}_{|\phi\rangle, \tau}^M)$ is the set of the subsets of $\mathcal{K}_{|\phi\rangle, \tau}^M$) defined by

$$\Pr \mathcal{B} = \sum_{R \in \mathcal{B}} \Pr\{R\}$$

for every $\mathcal{B} \subseteq \mathcal{K}_{|\phi\rangle, \tau}^M$.

By $\mathcal{K}[k]_{|\phi\rangle, \tau}^M$ we shall denote the set of the finite τ -observed runs of length k of M on $|\phi\rangle$, with the measure Pr on its subsets (see Definition 21).

It is immediate to observe that the set $\mathcal{K}_{|\phi\rangle, \tau}^M$ naturally defines an infinite tree labelled with q -configurations, where each infinite path starting from the root $|\phi\rangle$ corresponds to a τ -observed run in $\mathcal{K}_{|\phi\rangle, \tau}^M$.

Lemma 1. Given $R_1, R_2 \in \mathcal{K}_{|\phi\rangle, \tau}^M$ with $R_1 = \{\phi_{1,i}\}_{i \in \mathbb{N}} \neq \{\phi_{2,i}\}_{i \in \mathbb{N}} = R_2$, there is $k \geq 0$ s.t.

1. $\phi_{1,i} = \phi_{2,i}$ for $i \leq \tau(k)$, that is, $R_1[\tau(k)] = R_2[\tau(k)]$;
2. for $i > \tau(k)$, the q -configurations $\phi_{1,i} \neq \phi_{2,i}$ are in two orthonormal subspaces generated by two distinct subsets of \mathfrak{C} .

Proof. Let $R_1[h] = R_2[h]$ be the longest common prefix of R_1 and R_2 . Because they both starts with $|\phi\rangle$, such a prefix is not empty; moreover, by the definition of τ -observed run, it is readily seen that $h = \tau(k)$, for some k .

Let us now prove item 2 of Lemma 1. If we take $\mathcal{B}_{a,j} = \{C \in \mathfrak{C} \mid |C\rangle \in |\psi_{a,h+1+j}\rangle\}$, for $a = 1, 2$ and $j \in \mathbb{N}$, we need to prove that $\mathcal{B}_{1,j} \cap \mathcal{B}_{2,j} = \emptyset$, for every j . By construction, $\phi_{1,h+1} \neq \phi_{2,h+1}$ and, for $a = 1, 2$, $\phi_h \downarrow_{x_a} \phi_{a,h+1}$, for some x_a , with $\phi_h = \phi_{a,h}$. Moreover, $\mathcal{B}_{1,0} \cap \mathcal{B}_{2,0} = \emptyset$ (see Remark 8), at least one of the two observed values x_1, x_2 is not \perp , and one of the two q -configurations $|\phi_{1,h+1}\rangle, |\phi_{2,h+1}\rangle$ is a final base q -configuration. W.l.o.g., let us assume that $x_1 \in \mathbb{N}$ and let $|\phi_{1,h+1}\rangle = e|C, n\rangle$, for some final plain configuration C and some $n \in \mathbb{N}$. Subsequently, $|\phi_{1,h+1+j}\rangle = eU^j|C, n\rangle = e|C, n+j\rangle$ (see Remark 9) and $\mathcal{B}_{1,j} = \{|C, n+j\rangle\}$, for $j \in \mathbb{N}$. Thus, to prove the assertion, it suffices to show that $|C, n+j\rangle \notin \mathcal{B}_{2,j}$, for $j \in \mathbb{N}$. Let us distinguish two cases:

1. $x_2 \in \mathbb{N}$ and $|\psi_{2,h+1}\rangle = e'|D, m\rangle$, where D is a final plain configuration and $m \in \mathbb{N}$. As already seen for $|\psi_{1,h+1+j}\rangle$, for $j \in \mathbb{N}$, we have $|\psi_{2,h+1+j}\rangle = e'U^j|D, m\rangle = e'|D, m+j\rangle$ and $\mathcal{B}_{2,j} = \{|D, m+j\rangle\}$. Therefore, $|C, n+j\rangle \notin \mathcal{B}_{2,j}$, since $|C, n+j\rangle \neq |D, m+j\rangle$, for $|C, n\rangle \neq |D, m\rangle$ by construction.
2. $|\psi_{2,h+1}\rangle \in \ell^2(\mathfrak{C} \setminus \mathfrak{C}^{fin})$. Let $|D, m\rangle \in \mathcal{B}_{2,j} \cap \mathfrak{C}^{fin}$, where D is a plain configuration and $m \in \mathbb{N}$. By induction on j , it is readily seen that $m < j \leq n+j$. Thus, $|D, m\rangle \neq |C, n+j\rangle$.

□

Lemma 2. Let $K_{|\phi\rangle}^M = \{\phi_i\}_{i \in \mathbb{N}}$ be the computation of the QTM M on the initial q -configuration $|\phi\rangle$ and $\mathcal{K}_{|\phi\rangle, \tau}^M$ be the τ -observed computation on the same initial configuration. For every $k \in \mathbb{N}$, we have that

$$|\phi_k\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\text{Pr}\{R\}} |\psi_R\rangle$$

where $|\psi_R\rangle$ is the last q -configuration of the finite run R of length k .

Proof. By definition, $\phi = \phi_0$ and $R = \{\phi\}$ with $\text{Pr}\{R\} = 1$, and $\psi_R = \phi$ is the only run of length 0 in $\mathcal{K}_{|\phi\rangle, \tau}^M$. Therefore, the assertion trivially holds for $k = 0$.

Let us then prove the assertion by induction on k . By definition and the induction hypothesis

$$|\phi_{k+1}\rangle = U_M |\phi_k\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\text{Pr}\{R\}} U_M |\psi_R\rangle$$

We have two possibilities:

1. $k \neq \tau(i)$, for any i . In this case, there is a bijection between the runs of length k and those of length $k+1$, since each run $R' \in \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M$ is obtained from a run $R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M$ with last

q-configuration $|\psi_R\rangle$, by appending to R the q-configuration $|\psi_{R'}\rangle = U_M|\psi_R\rangle$. Moreover, because, by definition, $\Pr\{R'\} = \Pr\{R\}$, we can conclude that

$$|\phi_{k+1}\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} U_M|\psi_R\rangle = \sum_{R' \in \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R'\}} |\psi_{R'}\rangle$$

2. $k = \tau(i)$, for some i . In this case, every $R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M$ with last q-configuration $|\psi_R\rangle$ generates a run R' of length $k + 1$ for every output observation $|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle$, where R' is obtained by appending $|\psi_{R'}\rangle$ to R . Therefore, let $R = \{|\psi_i\rangle\}_{i \leq k} \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M$ and

$$\mathcal{B}_R = \{\{|\psi_i\rangle\}_{i \leq k+1} \mid |\psi_k\rangle \downarrow_x |\psi_{k+1}\rangle\}$$

By applying Definition 17, we easily check that

$$U_M|\psi_R\rangle = \sum_{R' \in \mathcal{B}_R} \sqrt{\Pr\{|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle\}} |\psi_{R'}\rangle$$

Thus, by substitution, and $\Pr\{R\} \Pr\{|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle\} = \Pr\{R'\}$

$$\begin{aligned} |\phi_{k+1}\rangle &= \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} U_M|\psi_R\rangle \\ &= \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sum_{R' \in \mathcal{B}_R} \sqrt{\Pr\{R\}} \sqrt{\Pr\{|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle\}} |\psi_{R'}\rangle \\ &= \sum_{R' \in \bigcup_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \mathcal{B}_R} \sqrt{\Pr\{R'\}} |\psi_{R'}\rangle \\ &= \sum_{R' \in \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R'\}} |\psi_{R'}\rangle \end{aligned}$$

since $\bigcup_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \mathcal{B}_R = \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M$.

□

We are finally in the position to prove that our observation protocol is compatible with the probability distributions that we defined as computed output of a QTM computation.

Theorem 4. Let $K_{|\phi\rangle}^M = \{\phi_i\}_{i \in \mathbb{N}}$ be the computation of the QTM M on the initial q-configuration $|\phi\rangle$ and $\mathcal{K}_{|\phi\rangle, \tau}^M$ be the τ -observed computation on the same initial configuration. For every $n \in \mathbb{N}$:

1. $\mathbf{P}_{|\phi_k\rangle}(n) = \Pr\{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M \mid R \downarrow_n\}$, for $k = \tau(i) + 1$ and $i \in \mathbb{N}$; and,
2. $\mathbf{P}_{K_{|\phi\rangle}^M}(n) = \Pr\{R \in \mathcal{K}_{|\phi\rangle, \tau}^M \mid R \downarrow_n\}$.

Proof. Let us start with the first item. By Lemma 2, we know that

$$|\phi_k\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} |\psi_R\rangle,$$

where $|\psi_R\rangle$ is the last q-configuration of R . Because $k = \tau(i) + 1$, for some i , we also know that either $\psi_R \in \mathcal{C} \setminus \mathcal{C}^{fin}$ or $|\psi_R\rangle = u_R|C_R\rangle$ with $C_R \in \mathcal{C}^{fin}$ and $|u_R| = 1$. Therefore,

$$\mathbf{P}_{|\phi_k\rangle}(n) = \left\| \sum_{R \in \mathcal{B}[k,n]} \sqrt{\Pr\{R\}} u_R |C_R\rangle \right\|^2$$

where

$$\begin{aligned} \mathcal{B}[k,n] &= \{R \in \mathcal{K}[k]_{|\phi\rangle,\tau}^M \mid R \downarrow_n\} \\ &= \{R \in \mathcal{K}[k]_{|\phi\rangle,\tau}^M \mid |\psi_R\rangle = u_R|C_R\rangle \text{ with } \mathbf{val}[C_R] = n\} \end{aligned}$$

By Lemma 1, we know that for every $R_1, R_2 \in \mathcal{B}[k,n]$, we have $|C_{R_1}\rangle \neq |C_{R_2}\rangle$. Therefore

$$\mathbf{P}_{|\phi_k\rangle}(n) = \sum_{R \in \mathcal{B}[k,n]} \Pr\{R\} |u_R|^2 = \sum_{R \in \mathcal{B}[k,n]} \Pr\{R\} = \Pr \mathcal{B}[k,n]$$

since $|u_R| = 1$. Which concludes the proof of the first item of the assertion.

In order to prove the second item, let $\mathcal{B}[\omega,n] = \{R \in \mathcal{K}_{|\phi\rangle,\tau}^M \mid R \downarrow_n\}$. We have that

$$\begin{aligned} \Pr\{R \in \mathcal{K}_{|\phi\rangle,\tau}^M \mid R \downarrow_n\} &= \sum_{R \in \mathcal{B}[\omega,n]} \Pr\{R\} = \lim_{i \rightarrow \infty} \sum_{\substack{R \in \mathcal{B}[\omega,n] \\ R[\tau(i)+1] \downarrow_n}} \Pr\{R\} \\ (*) &= \lim_{i \rightarrow \infty} \sum_{\substack{R \in \mathcal{B}[\omega,n] \\ R[\tau(i)+1] \downarrow_n}} \Pr\{R[\tau(i)+1]\} \\ (**) &= \lim_{i \rightarrow \infty} \sum_{R' \in \mathcal{B}[\tau(i)+1,n]} \Pr\{R'\} = \lim_{i \rightarrow \infty} \Pr \mathcal{B}[\tau(i)+1,n] \end{aligned}$$

since: $(*) \Pr\{R\} = \Pr\{R[\tau(i)+1]\}$, when $R[\tau(i)+1] \downarrow_n$ (see Remark 10); $(**)$, there is a bijection between $\mathcal{B}[\tau(i)+1,n]$ and $\{R \in \mathcal{B}[\omega,n] \mid R[\tau(i)+1] \downarrow_n\}$ (see Remark 9) mapping every $R' \in \mathcal{B}[\tau(i)+1,n]$ with last q-configuration $u|C\rangle$ into $R = \{\psi_j\}_{j \in \mathbb{N}} \in \{R \in \mathcal{B}[\omega,n] \mid R[\tau(i)+1] \downarrow_n\}$ s.t $R' = R[\tau(i)+1]$ and $|\psi_j\rangle = u U_M^{j-\tau(i)-1}|C\rangle$ for $j > \tau(i)$.

Therefore, by the (already proved) first item of the assertion

$$\Pr\{R \in \mathcal{K}_{|\phi\rangle,\tau}^M \mid R \downarrow_n\} = \lim_{i \rightarrow \infty} \Pr \mathcal{B}[\tau(i)+1,n] = \lim_{i \rightarrow \infty} \mathbf{P}_{|\phi_{\tau(i)+1}\rangle}(n) = \mathbf{P}_{K_{|\phi\rangle}^M}(n)$$

□

The above theorem shows that our protocol is not only realistic (unlike those of B&V and Deutsch), but also compatible with the theoretical probabilities defined for configurations in superposition. In particular, the theorem shows how it is possible to make measurements in a totally free way, without destroying the expressive power of quantum parallelism, i.e., without destroying the ability of a QTM to use quantum interference.

5. A Comparison With Bernstein and Vazirani's QTMs: Part 2

Theorem 2 has shown that our QTMs generalise B&V-QTMs, since the computation of any B&V-QTM can be simulated by a corresponding QTM with the same transition function (up to transitions entering/leaving the initial/final state). The general framework, however, is substantially modified, and the “same” machine behaves in different ways in the two approaches. In this section, we give two simple examples of this.

5.1. QTM Transition Graphs

Let us represent a QTM M by means of a transition graph (a directed graph) whose nodes are the states of M , and whose arrows give its transition function (see Figure 1). Namely, if $\delta_0(p, a)(q, b, d) = x \neq 0$, the graph of M contains an arrow from the node of p to the node of q , labelled by the tuple $(a \mapsto b, d, x)$. Every non-target node has at least one outgoing edge labelled by such a tuple, for any symbol a of the tape alphabet.

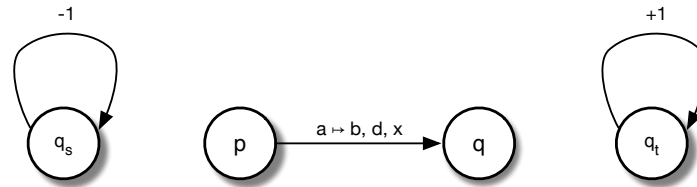


Figure 1. Transitions of a QTM.

In addition to the arrows of δ_0 , to represent the looping transitions of target nodes, the graph contains a self-loop labelled by $+1$ on any target node q_t : to denote the fact that the only transition of any target configuration is the one that increases the counter by 1, without modifying the rest of the configuration. Dually, every source node q_s has a self-loop labelled by -1 . The self-loop of q_s is not its only outgoing arrow, since the corresponding counter decreasing transition applies to source configurations with a counter greater than 0 only; indeed, when the counter of a source configuration reaches the 0, the δ_0 transition function applies. On the other hand, the self-loop is the only incoming arrow of any source state q_s , because no transition from another state can enter into it. The situation is dual for target states, for which the self-loop is the only outgoing arrow.

See Figure 4 for an example of source and target nodes of a transition graph: the state s and q_0 (which is also initial) are source states; the states p and q_f (which is also final) are target states.

5.2. A Classical Reversible TM With Quantum Behaviour

In Figure 2, we give the transition graph of a B&V-QTM corresponding to a reversible TM: any reversible TM M can be transformed into a B&V-QTM by assuming that the weight of any transition of M is 1, and by adding a back-transition from the final state to the initial state.

In the example presented in Figure 2, q_0 is the initial state and q_f is the final one, and they are connected by an arrow $(a \mapsto a, R, 1)$ from q_f to q_0 , where $a \in \{\square, 1\}$. When started on an initial tape containing $\underline{n+1}$ (that is, $n+2$ symbols 1, see Section 2), the machine M erases two 1s from the tape. When the initial tape contains instead $\underline{0}$, M loops indefinitely on the middle state q , after erasing the unique symbol 1 on the tape. Summing up (remembering that for the computed output we just count the number of 1s on the tape, see Definition 12), M computes the predecessor $n-1$ (with probability 1), when fed with a representation of $n > 0$, while it diverges (with probability 1) for $n = 0$. However, if we feed M with a non classical input as $|\psi\rangle = \frac{1}{\sqrt{2}}|\underline{0}\rangle + \frac{1}{\sqrt{2}}|\underline{2}\rangle$, then M fails to give an answer according to B&V's framework, since it reaches a q -configuration in which final and non-final base configurations superpose.

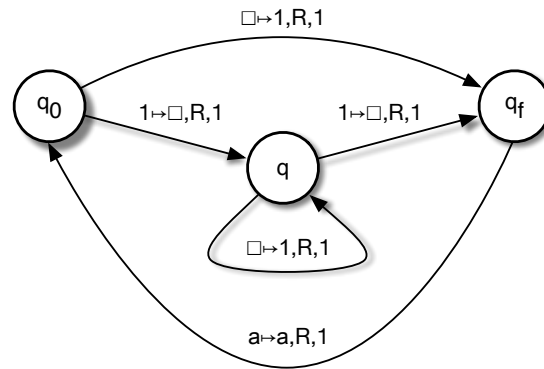


Figure 2. A reversible TM as a quantum Turing Machines (QTM) à la Bernstein and Vazirani.

To transform M into a QTM according to our formalism (see Theorem 2), it suffices to replace the arrow from q_f to q_0 by two self-loops: one (labelled -1) on the initial state q_0 , and one (labelled $+1$) on the final state q_f . We obtain then the QTM in Figure 3. By the definition of computed output, we can see that $M_{\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|2\rangle} \rightarrow \{1 \mapsto 1/2; n \mapsto 0, \text{ if } n \neq 1\}$; namely, with probability $1/2$ the QTM halts with computed output 1 ; while with probability $1/2$ it diverges.

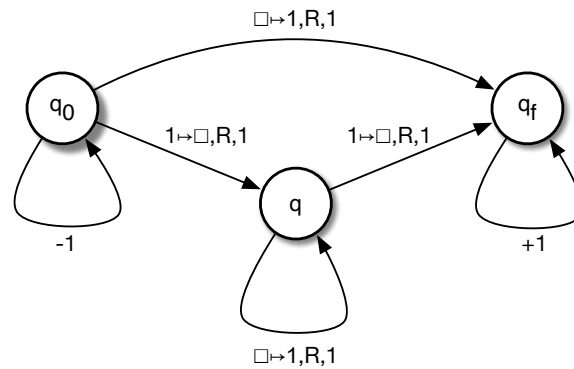


Figure 3. A reversible TM as a QTM.

5.3. A PD Obtained as a Limit

The example presented in Figure 4 shows a QTM that produces a PD only as an infinite limit. The tape alphabet of the machine M is $\Sigma = \{\$, 1, \square\}$, the set of its source states is $Q_s = \{q_0, s\}$, the set of its target states is $Q_t = \{q_f, p\}$, the state q_0 is initial, the state q_f is final. In the figure, the symbol a stands for any symbol of the alphabet but $\$$, which is, $a \in \{1, \square\}$.

The machine M applies properly on initial configurations $\langle \lambda, q_0, \$n, 0, 0 \rangle$, whose corresponding base vector will be denoted by $|\$n\rangle$ (to simplify the definition of the machine, we use a slight variant of the encoding given in Section 3.5 in which the sequence n coding the number n is preceded by a $\$$, which is indeed the only $\$$ on the tape). On such inputs, after moving from the initial state q_0 to q_1 , M either ends up in the final state q_f with probability $1/2$, or it loops on q_1 with probability $1/2$. We remark the source state s and the target state p : such states do not play any role when the machine computes on $\$n$, since none of them can be reached. Nevertheless, they must be added to deal with error situations, and to satisfy the local unitary conditions, in order to obtain a proper unitary time evolution for M (see Remark 1).

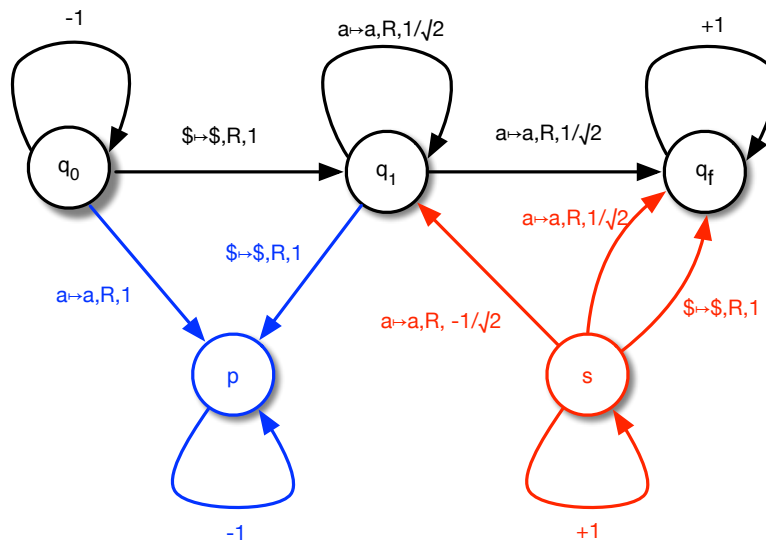


Figure 4. A QTM computing the successor function (the identity w.r.t. the tape) as a limit.

A simple calculation shows that $M_{|\$n\rangle} \rightarrow \mathbf{P}$, with $\mathbf{P} = \{n + 1 \mapsto 1; m \mapsto 0, \text{ if } m \neq n + 1\}$; namely, on the input $\$n$, M computes with probability 1 the successor $n + 1$ (the machine is indeed the identity w.r.t. the tape, since no transition of the machine changes any symbol on the tape; however, since n is a sequence of $n + 1$ symbols 1, the computation leaves all these symbols 1 on the tape, leading then to a computed output of $n + 1$). We stress that the PD \mathbf{P} is obtained as a limit, since for every $j \in \mathbb{N}$, $\mathbf{P}_{U^j|\$n\rangle}$ is a PPD s.t. $\mathbf{P}_{U^j|\$n\rangle}(n + 1) < 1$ and $\mathbf{P}_{U^j|\$n\rangle}(m) = 0$, for $m \neq n + 1$. Of course, this does not mean that we have to wait an infinite time to readback the result! A correct way to interpret this fact is that, for each $n \in \mathbb{N}$, and each $\epsilon \in (0, 1/2]$, there exists $k \in \mathbb{N}$ s.t. for every $j > k$, $1 - \epsilon < \mathbf{P}_{U^j|\$n\rangle}(n + 1) \leq 1$.

6. Related Works

After having analysed at the length the notions of B&V and Deutsch, we discuss in this section some other papers related to our work.

6.1. On Quantum Extensions of Turing Machines, And of Related Complete Formalisms

- Strictly following the B&V approach, Nishimura and Ozawa [17,24] study the relationship between QTMs and quantum circuits (extending previous results by Yao [25]). They show that there exists a perfect computational correspondence between QTMs and uniform, finitely generated families of quantum circuits. Such a correspondence preserves the quantum complexity classes EQP, BQP, and ZQP.
- Perdrix and Jorrand [26] proposes a new way to deal with quantum extensions of Turing Machines. The basic idea is reminiscent of the quantum-data/classical-control paradigm coined by Selinger [27,28]. In fact, in Perdrix QTM's, the only quantum component is the tape whereas the control is completely classical.
- Dal Lago, Masini, and Zorzi [3,29,30] extend the quantum-data/classical-control paradigm to a type free quantum λ -calculus that is proven to be in perfect correspondence with the QTMs of B&V. Following the ideas of the so-called Implicit Computational Complexity, the authors propose an alternative way to deal with the quantum classes EQP, BQP, and ZQP.

6.2. On the Readout Problem

The following papers address the problem of how to readout the result of a quantum computation. Because this is a key question in the definition of any quantum computing formalism, they deserve some deeper attention.

We recall, however, that our main interest is in QTMs as devices computing distributions of probability, and not functions over natural numbers.

- Myers [31] tries to show that it is not possible to define a truly quantum general computer. The article highlights how the B&V approach fails on truly quantum data. In fact, in such a case, it is impossible to guarantee the synchronous termination of all the computations in superposition. Consequently, the use of a termination bit spoils the quantum superposition of the computation. This defect was well known, and it is for this reason that B&V did not define a general notion of quantum computability, but rather a notion sufficient to solve—in a quantum way—only classical decision problems. Myers's criticism does not apply to our approach. Our QTMs are fully quantum, and they have an observational protocol of the result that does not depend on the synchronous termination of the computations in superposition.
- In an unpublished note, Kieu and Danos [32] claim that: "For halting, it is desirable of the dynamics to be able to store the output, which is finite in terms of qubit resources, invariantly (that is, unchanged under the unitary evolution) after some finite time when the desirable output has been computed". Unfortunately, it is not possible to enter into a truly invariant final quantum configuration—only a machine starting in a final configuration and computing the identity can accomplish this constraint. We overcome the problem by introducing a feasible (i.e., correct from a quantum point of view) notion of invariant, w.r.t. the readout, of final configurations. In this way, even if the final configuration changes, the output we read from that configuration does not change.
- In another unpublished note, Linden and Popescu [22] address the problem of how to readout the result of a general quantum computer. The authors write: "We explicitly demonstrate the difficulties that arise in a quantum computer when different branches of the computation halt at different, unknown, times", implicitly referring to the problems in extending the approach of B&V to general quantum inputs (see again [31], discussed above). In the first part of the work, the authors show that the problem cannot be solved by means of the so-called "ancilla". The ancilla is an additional information added to the main information encoded by a configuration of the quantum machine. The idea is that, once a final state is reached, the machine keeps modifying the ancilla only. The authors show that the ancilla approach destroys the quantum capabilities of quantum machines, since only classical computations can survive to this treatment of ancilla. Even if reminiscent of the ancilla, our approach is technically different—the problems addressed by Linden and Popescu do not apply—since we carefully tailor the space of the possible configurations of the machines, allowing for the ancilla to play a role only during its final and initial evolution.

In the second part of the work, the authors launch a strong attack against the use of termination bit, the solution originally proposed by Deutsch and successively refined by Ozawa [13]. The authors try to argue that the approach that was proposed by Deutsch/Ozawa cannot work. In fact, they show that, even if it is true that once the termination bit is set to 1, it remains firmly with such a value forever, any terminal configuration cannot be frozen, and keeps evolving according to the Hamiltonian of the system. Once again, our proposal does not have the defect depicted in the paper, because, far away to force a final configuration to remain stable, only the readout of a final configuration is stable in our approach.

- Hines [33] shows how to ensure simultaneous coherent halting, provided that termination is guaranteed for a restricted class of quantum algorithms. This kind of approach that is based on coherent halting is intentionally not followed in our paper. Indeed, as previously remarked,

we are interested in treating systems which includes, as a particular case, all classical computable functions—we cannot restrict to terminating computations.

- Miyadera and Ohya [34] discuss the notion of *probabilistic halting*. In particular, “[...] the notion of halting is still probabilistic. That is, a QTM with an input sometimes halts and sometimes does not halt. If one can not get rid of the possibility of such a probabilistic halting, one can not tell anything certain for one experiment since one can not say whether an event of halting or non-halting occurred with probability one or just by accident, say with probability 10^{-40} ”. Therefore, they wonder about the existence of any algorithm to decide whether or not a QTM probabilistic halts. With no surprise, they conclude that such an algorithm cannot exist. In fact, because the non-probabilistic halting of a QTM corresponds to the simultaneous halting of all the superposing branchings of its computation, an algorithm deciding the probabilistic halting would decide the simultaneous termination of two classical reversible machines (by combining them into a unique QTM), which is clearly an undecidable problem. In a sense, the question of probabilistic or non-probabilistic halting is irrelevant for our approach. In our QTMs, the result of a computation is defined as a limit, and any computation converges to some result. Accordingly, the only possible readouts that we can get are approximations of such a result. At the same time, because we show that a repeated-measures protocol can retrieve the distribution associated to the output of a computation, we can accept to say that our approach is probabilistic.

7. Conclusions and Further Work

We find surprising that, in the thirty years since [10], a theory of quantum computable functions did not develop, and that the main interest remained in QTMs as computing devices for classical problems/functions. This is in sharp contrast with the original (Feynman’s and Deutsch’s) aim to have a better computing simulation of the physical world.

As always in these foundational studies, we had to go back to the basics, and look for a notion of QTM general enough to encompass previous approaches. We started with pre Quantum TMs (Definition 1), and defined then QTMs as those pre QTMs whose time evolution operator is unitary (Definition 7). Following analogous results in the literature, we showed that unitarity of the evolution operator for a QTM M is equivalent to a set of constraints formulated on the transition function of M (Theorem 1). Our QTMs are able to simulate B&V-QTMs (Theorem 2), but, contrary to B&V-QTMs, they may be used to give meaning to some “infinite” computations. Indeed, in Section 3 we showed that each QTM M naturally defines a computable function from the sphere of radius 1 in $\ell^2(\mathcal{C}_M)$ (i.e., the superpositions of the configurations of M) to the set of (partial) probability distributions on the set of natural numbers. Such a definition is one of the main reasons for our definition of QTM, for it involves computations where some path “terminates”, while others do not. We remark that the definition implies monotonicity of quantum computations (Theorem 3), which is a consequence of the particular way final states are treated in order to defuse quantum interference, once such states are entered.

Particularly tricky is the way in which one reads a result from a quantum computation, which is the subject of Section 4, and, in particular, Section 4.5, which shows that our protocol is compatible with the theoretical probabilities defined for configurations in superposition (Theorem 4).

While several details of the proposed approach may well change during further study, we are confident the general picture may allow a fresh look to the problem of a quantum universal machine, to the termination of QTMs (for instance, characterising termination into the arithmetical hierarchy), and to the various degrees of partiality of quantum computable functions.

Author Contributions: All authors have contributed equally to the paper. All authors have read and agreed to the published version of the manuscript.

Funding: S. Martini received partial support from INdAM-GNSAGA.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Hilbert Spaces With Denumerable Basis

Definition A1 (Hilbert space of configurations). *Given a denumerable set \mathcal{B} , with $\ell^2(\mathcal{B})$ we shall denote the infinite dimensional Hilbert space defined as follow.*

The set of vectors in $\ell^2(\mathcal{B})$ is the set

$$\left\{ \phi \mid \phi : \mathcal{B} \rightarrow \mathbb{C}, \sum_{C \in \mathcal{B}} |\phi(C)|^2 < \infty \right\}$$

and equipped with:

1. An inner sum $+$: $\ell^2(\mathcal{B}) \times \ell^2(\mathcal{B}) \rightarrow \ell^2(\mathcal{B})$
defined by $(\phi + \psi)(C) = \phi(C) + \psi(C)$;
2. A multiplication by a scalar \cdot : $\mathbb{C} \times \ell^2(\mathcal{B}) \rightarrow \ell^2(\mathcal{B})$
defined by $(a \cdot \phi)(C) = a \cdot (\phi(C))$;
3. An inner product $\langle \cdot, \cdot \rangle$: $\ell^2(\mathcal{B}) \times \ell^2(\mathcal{B}) \rightarrow \mathbb{C}$
defined by $\langle \phi, \psi \rangle = \sum_{C \in \mathcal{B}} \phi(C)^* \psi(C)$ (observe that the condition $\sum_{C \in \mathcal{B}} |\phi(C)|^2 < \infty$ implies that $\sum_{C \in \mathcal{B}} \phi(C)^* \psi(C)$ converges for every pair of vectors).
4. The Euclidian norm is defined as $\|\phi\| = \sqrt{\langle \phi, \phi \rangle}$.

The Hilbert space $\ell^2 = \ell^2(\mathbb{N})$ is the standard Hilbert space of denumerable dimension—all the Hilbert spaces with denumerable dimension are isomorphic to it. ℓ_1^2 is the set of the vectors of ℓ^2 with unit norm.

Definition A2 (computational basis). *The set of functions*

$$\text{CB}(\mathcal{B}) = \{ |C\rangle : C \in \mathcal{B}, |C\rangle : \mathcal{B} \rightarrow \mathbb{C} \}$$

such that for each C

$$|C\rangle(D) = \begin{cases} 1 & \text{if } C = D \\ 0 & \text{if } C \neq D \end{cases}$$

is called computational basis of $\ell^2(\mathcal{B})$.

We can prove that [16]:

Theorem A1. *The set $\text{CB}(\mathcal{B})$ is a Hilbert basis of $\ell^2(\mathcal{B})$.*

Let us note that the inner product space $\text{span}(\text{CB}(\mathcal{B}))$ defined by:

$$\text{span}(\text{CB}(\mathcal{B})) = \left\{ \sum_{i=1}^n c_i S_i \mid c_i \in \mathbb{C}, S_i \in \text{CB}(\mathcal{B}), n \in \mathbb{N} \right\}.$$

is a proper inner product subspace of $\ell^2(\mathcal{B})$, but it is not an Hilbert Space (this means that $\text{CB}(\mathcal{B})$ is not an Hamel basis of $\ell^2(\mathcal{B})$).

The completion of $\text{span}(\text{CB}(\mathcal{B}))$ is a space isomorphic to $\ell^2(\mathcal{B})$.

By means of a standard result in functional analysis we have:

Theorem A2.

1. $\text{span}(\text{CB}(\mathcal{B}))$ is a dense subspace of $\ell^2(\mathcal{B})$;
2. $\ell^2(\mathcal{B})$ is the (unique! up to isomorphism) completion of $\text{span}(\text{CB}(\mathcal{B}))$.

Definition A3. Let \mathcal{V} be a complex inner product space, a linear application $U : \mathcal{V} \rightarrow \mathcal{V}$ is called an isometry if $\langle Ux, Uy \rangle = \langle x, y \rangle$, for each $x, y \in \mathcal{V}$; moreover if U is also surjective, then it is called unitary.

Since an isometry is injective, a unitary operator is invertible, and moreover, its inverse is also unitary.

Definition A4. Let \mathcal{V} be a complex inner product vector space, a linear application $L : \mathcal{V} \rightarrow \mathcal{V}$ is called bounded if $\exists c > 0 \forall x \|Lx\| \leq c\|x\|$.

Theorem A3. Let \mathcal{V} be a complex inner product vectorial space, for each bounded application $U : \mathcal{V} \rightarrow \mathcal{V}$ there is one and only one bounded application $U^* : \mathcal{V} \rightarrow \mathcal{V}$ s.t. $\langle x, Uy \rangle = \langle U^*x, y \rangle$. We say that U^* is the adjoint of U .

It is easy to show that if U is a bounded application, then U is unitary iff U is invertible and $U^* = U^{-1}$.

Theorem A4. Each bounded unitary operator U in $\text{span}(\text{CB}(\mathcal{B}))$ has an unique continuous extension in $\ell^2(\mathcal{B})$ [12].

Appendix A.1. Dirac Notation

We conclude this brief digest on Hilbert spaces, by a synopsis of the so-called Dirac notation, extensively used in the paper.

mathematical notion	Dirac notation
inner product $\langle \phi, \psi \rangle$	$\langle \phi \psi \rangle$
vector ϕ	$ \phi\rangle$
dual of vector ϕ i.e., the linear application d_ϕ defined as $d_\phi(\psi) = \langle \phi, \psi \rangle$	$\langle \phi $ note that $\langle \phi \psi \rangle = \langle \phi (\psi\rangle)$

Let L be a linear application, with $\langle \phi | L | \psi \rangle$ we denote $\langle \phi | L \psi \rangle$.

Appendix B. Implementation of the Counter

A TM purist might argue that the counter adds to the machine a device with a denumerable set of symbols, or with a denumerable set of states, which is not in the spirit of the finite representability of TMs. However, it is an easy exercise to implement the counter directly into the QTM, and in the following we shall briefly describe two ways to do it. Nevertheless, we stress that none of these implementations can be seen as natural or standard, and that, on the other hand, one could completely ignore the problem, assuming to add a clock to implement the counter. For these reasons, in the paper, we have preferred to give the more abstract solution, instead of any more concrete implementation.

Appendix B.1. Extra Symbols

A first possibility, that we followed in a previous version of the paper, is to duplicate the symbol alphabet Σ by adding a set of *extra tape symbols* $\bar{\Sigma} = \{\bar{a} \mid a \in \Sigma\}$: a new symbol \bar{a} , for any $a \in \Sigma$ (including the blank \square). In this way, when in a final state, the machine replaces any symbol a with the corresponding extra symbol \bar{a} , and moves to the right. Dually, when in a source state, if the current cell contains an extra symbol \bar{a} , the machine replaces the current symbol with the corresponding symbol a and moves to the right; otherwise, when the current symbol is $a \in \Sigma$, it behaves as specified by the main transition function δ_0 . The legal configurations are then restricted to three possible cases:

1. $\langle \alpha, q, \beta, i \rangle$

2. $\langle \alpha \bar{\gamma}, q_t, \beta, i \rangle$
3. $\langle \alpha, q_s, \bar{\gamma} \beta, i \rangle$

where: $\alpha \beta \gamma \in \Sigma^*$, $\bar{\gamma} \in \bar{\Sigma}^*$ is the sequence of extra symbols obtained by replacing any symbol a of γ with the corresponding extra symbol \bar{a} ; q is any state; q_t is a target state; q_s is a source state. It is readily seen that, to obtain an isomorphism between QTMs with extra symbols and QTMs with counters, it suffices to take the following bijection of configurations (where we use the same symbols as above):

1. $\langle \alpha, q, \beta, i \rangle \mapsto \langle \alpha, q, \beta, i, 0 \rangle$
2. $\langle \alpha \bar{\gamma}, q_t, \beta, i \rangle \mapsto \langle \alpha, q_t, \gamma \beta, i - |\gamma|, |\gamma| \rangle$
3. $\langle \alpha, q_s, \bar{\gamma} \beta, i \rangle \mapsto \langle \alpha \gamma, q_s, \beta, i + |\gamma|, |\gamma| \rangle$

where $|\gamma|$ denotes the length of γ .

Appendix B.2. Additional Counter Tape

Another possibility is to add a second tape to the machine. The alphabet of this counter tape contains only one symbol $*$, in addition to the blank \square corresponding to the empty cell. A counter containing a value of n corresponds then to a tape with n symbols $*$, see Figure A1.

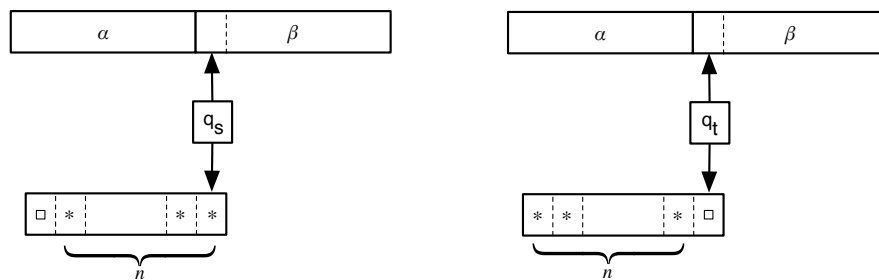


Figure A1. Implementation of the counter by means of a second tape with only one non blank symbol $*$.

Adding/subtracting 1 to the counter corresponds to write/delete a $*$ symbol. To implement these operations by a single step, it suffices that:

1. When the machine is in a source state q_s , the counter head is on the rightmost $*$ of the counter tape, if $n > 0$, or on an empty cell, if $n = 0$. If the current counter symbol is a $*$, any transition in the state q_s replaces such a $*$ with a \square , and moves the counter head to the left, until the current counter symbol becomes a \square , in which case the machine starts its main evolution.
2. When the machine is in a target state q_t , the counter head is on the first empty cell of the counter tape to the right of the sequence of $*$. Any transition in the state q_t replaces then the \square in the current counter cell with a $*$, and moves the counter head to the right.
3. When the state is neither a source nor a target state, the counter tape is empty, and any transition leaves the counter tape unchanged.

References

1. Backus, J.W. The Syntax and Semantics of the Proposed International Algebraic Language of the Zurich ACM-GAMM Conference. In Proceedings of the 1st International Conference on Information Processing, UNESCO, Paris, France, 15–20 June 1959; pp. 125–132.
2. McCarthy, J. A Basis for a Mathematical Theory of Computation. In *Computer Programming and Formal Systems*; Studies in Logic and the Foundations of Mathematics Series; Braffort, P., Hirschberg, D., Eds.; North-Holland Publishing Company, Elsevier B.V.: Amsterdam, The Netherlands, 1963; Volume 35, pp. 33–70.
3. Dal Lago, U.; Masini, A.; Zorzi, M. Confluence Results for a Quantum Lambda Calculus with Measurements. *Electron. Notes Theor. Comput. Sci.* **2011**, *270*, 251–261. [\[CrossRef\]](#)

4. Dal Lago, U.; Zorzi, M. Wave-Style Token Machines and Quantum Lambda Calculi. In Proceedings of the Third International Workshop on Linearity, LINEARITY 2014, Vienna, Austria, 13 July 2014; Alves, S., Cervesato, I., Eds.; Open Publishing Association: Waterloo, Australia, 2014; Volume 176, pp. 64–78. [\[CrossRef\]](#)
5. Zorzi, M. On quantum lambda calculi: A foundational perspective. *Math. Struct. Comput. Sci.* **2016**, *26*, 1107–1195. [\[CrossRef\]](#)
6. Viganò, L.; Volpe, M.; Zorzi, M. A branching distributed temporal logic for reasoning about entanglement-free quantum state transformations. *Inf. Comput.* **2017**, *255*, 311–333. [\[CrossRef\]](#)
7. Paolini, L.; Roversi, L.; Zorzi, M. Quantum programming made easy. In Proceedings of the Joint International Workshop on Linearity & Trends in Linear Logic and Applications, Linearity-TLLA@FLoC 2018, Oxford, UK, 7–8 July 2018; Ehrhard, T., Fernández, M., de Paiva, V., de Falco, L.T., Eds.; Open Publishing Association: Waterloo, Australia, 2018; Volume 292, pp. 133–147. [\[CrossRef\]](#)
8. Masini, A.; Zorzi, M. A Logic for Quantum Register Measurements. *Axioms* **2019**, *8*, 25. [\[CrossRef\]](#)
9. Paolini, L.; Piccolo, M.; Zorzi, M. QPCF: Higher-Order Languages and Quantum Circuits. *J. Autom. Reason.* **2019**, *63*, 941–966. [\[CrossRef\]](#)
10. Deutsch, D. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc. R. Soc. Lond. Ser. A* **1985**, *A400*, 97–117.
11. Benioff, P. The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *J. Stat. Phys.* **1980**, *22*, 563–591. [\[CrossRef\]](#)
12. Bernstein, E.; Vazirani, U. Quantum Complexity Theory. *SIAM J. Comput.* **1997**, *26*, 1411–1473. [\[CrossRef\]](#)
13. Ozawa, M. Quantum Nondemolition Monitoring of Universal Quantum Computers. *Phys. Rev. Lett.* **1998**, *80*, 631–634. [\[CrossRef\]](#)
14. Davis, M. *Computability and Unsolvability*; McGraw-Hill Series in Information Processing and Computers; McGraw-Hill Book Co. Inc.: New York, NY, USA, 1958.
15. Conway, J.B. *A Course in Functional Analysis*, 2nd ed.; Graduate Texts in Mathematics Series; Springer: New York, NY, USA, 1990; Volume 96, xvi+399p.
16. Roman, S. *Advanced Linear Algebra*, 3rd ed.; Graduate Texts in Mathematics Series; Springer: New York, NY, USA, 2008; Volume 135, xviii+522p.
17. Nishimura, H.; Ozawa, M. Computational complexity of uniform quantum circuit families and quantum Turing machines. *Theor. Comput. Sci.* **2002**, *276*, 147–181. [\[CrossRef\]](#)
18. Ozawa, M.; Nishimura, H. Local transition functions of quantum Turing machines. *Theor. Inform. Appl.* **2000**, *34*, 379–402. [\[CrossRef\]](#)
19. Bennett, C.H. Logical reversibility of computation. *IBM J. Res. Dev.* **1973**, *17*, 525–532. [\[CrossRef\]](#)
20. Kaye, P.; Laflamme, R.; Mosca, M. *An Introduction to Quantum Computing*; Oxford University Press: Oxford, UK, 2007; xii+274p.
21. Isham, C.J. *Lectures on Quantum Theory. Mathematical and Structural Foundations*; Imperial College Press: London, UK, 1995; xii+220p.
22. Linden, N.; Popescu, S. The Halting Problem for Quantum Computers. *arXiv* **1998**, arXiv:quant-ph/9806054.
23. Ozawa, M. Halting of Quantum Turing Machines. In *Unconventional Models of Computation*; Springer: Berlin/Heidelberg, Germany 2002; pp. 58–65.
24. Nishimura, H.; Ozawa, M. Perfect computational equivalence between quantum Turing machines and finitely generated uniform quantum circuit families. *Quantum Inf. Process.* **2009**, *8*, 13–24. [\[CrossRef\]](#)
25. Yao, A. Quantum Circuit Complexity. In Proceedings of the 34th Annual Symposium on Foundations of Computer Science, Palo Alto, CA, USA, 3–5 November 1993; IEEE Press: Los Alamitos, CA, USA, 1993; pp. 352–360.
26. Perdrix, S.; Jorrand, P. Classically controlled quantum computation. *Math. Struct. Comput. Sci.* **2006**, *16*, 601–620. [\[CrossRef\]](#)
27. Selinger, P. Towards a Quantum Programming Language. *Math. Struct. Comput. Sci.* **2004**, *14*, 527–586. [\[CrossRef\]](#)
28. Selinger, P.; Valiron, B. A lambda calculus for quantum computation with classical control. *Math. Struct. Comput. Sci.* **2006**, *16*, 527–552. [\[CrossRef\]](#)
29. Dal Lago, U.; Masini, A.; Zorzi, M. On a Measurement-Free Quantum Lambda Calculus with Classical Control. *Math. Struct. Comput. Sci.* **2009**, *19*, 297–335. [\[CrossRef\]](#)

30. Dal Lago, U.; Masini, A.; Zorzi, M. Quantum implicit computational complexity. *Theor. Comput. Sci.* **2010**, *411*, 377–409. [[CrossRef](#)]
31. Myers, J.M. Can a universal quantum computer be fully quantum? *Phys. Rev. Lett.* **1997**, *78*, 1823–1824. [[CrossRef](#)]
32. Kieu, T.D.; Danos, M. The Halting problem for universal quantum computers. *arXiv* **1998**, arXiv:quant-ph/9811001.
33. Hines, P. Quantum circuit oracles for Abstract Machine computations. *Theor. Comput. Sci.* **2010**, *411*, 1501–1520. [[CrossRef](#)]
34. Miyadera, T.; Ohya, M. On Halting Process of Quantum Turing Machine. *Open Syst. Inf. Dyn.* **2005**, *12*, 261–264. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).