Mirrors and Memory in Quantum Automata*

Carla Piazza and Riccardo Romanello

Dept. of Mathematics, Computer Science and Physics, University of Udine, Italy
{riccardo.romanello,carla.piazza}@uniud.it

Abstract. In this paper we start from the simplest form of Quantum Finite Automata (QFAs), namely Measure-Once QFAs with cut-point. First we elaborate on a variant of their semantics that can be obtained through a shift from the Schrödinger to the Heisenberg picture of Quantum Mechanics. In the Schrödinger picture states evolve in time while observables remain constant, while in the Heisenberg one states are constant and observables evolve. Interestingly, in the case of a QFA such shift reverts time-evolution. However, the equivalence of the two pictures over the class of QFAs holds thanks to the closure of the class with respect to language mirroring. Since the expressive power of such class of automata remains limited to infinite languages, we then consider their extension with bounded (multi-letter QFAs) and unbounded memory. Unfortunately, while bounded memory enhances the expressive power, the unbounded memory approach does not behave as one would expect.

Keywords: Quantum Automata, Heisenberg Picture, Language Mirroring, Memory in Quantum Automata.

Introduction

Deterministic and Nondeterministic Finite State Automata (DFA/NFA) are the building blocks of classical computation. They are the models at the basis of Verification Techniques such as Temporal Logic Model Checking [15].

A shift to their probabilistic and stochastic counter-parts is necessary whenever the evolution of the computation depends on probabilities and rates. In this context models such as Probabilistic/Stochastic Automata, Discrete/Continuous Time Markov Chains, and Probabilistic/Stochastic Process Algebra have been described (e.g., [20,21,24]). Their formal analysis involves performance metrics, behavioural equivalences, and extensions of temporal logics.

A currently emerging field in the context of Quantitative Computation and Performances Evaluation is Quantum Computation, where again extensions of automata, Markov chains, and temporal logics constitute a starting point for understanding properties of the computations (e.g., [19,17,3]).

* This work is partially supported by PRIN MUR project Noninterference and Reversibility Analysis in Private Blockchains (NiRvAna) - 20202FCJM and by GNCS INdAM project LESLIE.

Even though Quantum Automata have been studied since the end of the nineties, still today there is not a unique widely accepted definition of Quantum Finite Automata (QFAs). Moore and Crutchfield [27] introduced the idea of General Quantum Automata and characterized the properties of Quantum Regular Languages. The model they introduced was named *Measure-Once Quantum Finite Automata* (MO-QFAs) because the result can be observed (measured) only when the read of the input string has terminated. In the same years, Kondacs and Watrous in [23] introduced a different model of QFAs in which measurements can be used at each step of the computation. For this reason, these are called *Measure-Many Quantum Finite Automata* (MM-QFAs). Moreover, similarly to what happens on probabilistic automata [32], a key role in the expressive power of such models is played by the acceptance condition. The two most adopted conditions are called *cut-point* and *bounded error*.

The expressive power of both MO-QFA and MM-QFA has been deeply investigated in [2,14,8]. The expressive power of MO-QFAs does not include all languages accepted by DFAs. As a consequence, different extensions have been considered. In [1] a model called *Latvian* QFAs was considered. Bertoni *et al* [9] introduced MO-QFAs *with control language* which are able to recognize regular languages with bounded error. The same behavior can be found in a formalism in which a MO-QFAs are used together with a classical set of states [31]. Another model that can at least recognize regular languages was presented in [29] were the concept of *Ancilla* qubits is used.

In this paper we are interested in the most simple of these models, e.g., MO-QFAs with cut-point acceptance condition. In the case of Quantum Circuits the principle of *deferred measurements* states that measurements can always be moved from an intermediate stage to the final step. This is not true in the case of Quantum Automata, since MO-QFAs and MM-QFAs are not equivalent. So, a Measure-Once condition is more in the spirit of a basic model. As for the acceptance condition, bounded error ensures the possibility of arbitrarily improving the precision. Consequently, it has been largely studied in the literature. However, it is not the "equivalent" of what happens in experimental disciplines such as biology and medicine, where cut-offs have to be arbitrarily chosen and no separation is guaranteed between positive and negative answers.

First, in this paper we analyse whether it is possible to increase the expressive power of MO-QFAs with cut-point without enriching their syntax, but simply moving to an alternative semantics. Such semantics from the point of view of physics is as natural as the one which has been considered in the literature so far. We are talking of a shift from the Schrödinger picture of Quantum Mechanics to the Heisenberg one. We will not obtain a positive answer in terms of increase of the expressive power, but our investigation provides a closure property of MO-QFAs with respect to mirror images which is new. In other terms, the mirror closure proves that not only each internal step of a MO-QFA is reversible, but its computation as a whole is. Such result was not granted because of the asymmetric use in MO-QFAs of final states and measurement.

As a second step, we are interested in considering another semantics for MO-QFAs. This is only inspired by the Heisenberg picture and at first sight it seems to provide an unbounded quantity of memory to the automata. In particular, at each point of the computation all the prefix that has been read so far is involved in the choice of the evolution. However, as it usually happens in the quantum realm, our intuition is cheated and such unbounded quantity of memory is less expressive than expected. Again, the path which leads us to such "negative" result is interesting by itself. We quantify the minimum amount of memory necessary for accepting finite languages and provide a pumping lemma for a class of QFAs which have been studied in the literature with bounded error, but not with cut-point [6,30].

The paper is organized as follows. In Section 1 we give a brief presentation of the notation and the basic concepts that are useful throughout the paper. In Section 2 we introduce MO-QFAs and we briefly survey the state of the art about their expressive power and realizations. These results will be useful in Section 3 where we define *Heisenberg Quantum Finite Automata* (HQFAs) and compare them with MO-QFAs. In Section 4 we study a class of *Heisenberg inspired* automata which we call *Unbounded Memory Quantum Automata* (UMQFAs) and we compare them with a *bounded memory* counter-part. The proofs of the main results of this paper can be found in the Appendix.

1 Preliminaries

1.1 Strings and Languages

An alphabet Σ is a set of symbols. We always refer to finite alphabets. A string $\mathbf{x} = x_1 x_2 \ldots x_m$ of length m over Σ is a finite sequence of symbols $x_i \in \Sigma$. The empty string ϵ is the only string of length 0. With Σ^i we indicate the set of all strings of length i over Σ, while $\Sigma^{\leq i} = \cup_{j=0}^{i} \Sigma^j$ is the set of all strings of length at most i. $\Sigma^* = \cup_{i \in \mathbb{N}} \Sigma^i$ is the set of all finite length strings we can build on Σ.

Given a string $\mathbf{x} = x_1 x_2 \ldots x_m$ we denote by $\overleftarrow{\mathbf{x}}$ its mirror image, i.e., the string $\overleftarrow{\mathbf{x}} = x_m x_{m-1} \ldots x_1$. Given an index $1 \leq j \leq n$ we denote by $\mathbf{x}_j$ the prefix of $\mathbf{x}$ from x_1 to x_{j-1}, i.e., $\mathbf{x}_j = x_1 x_2 \ldots x_{j-1}$. If $j = 1$, then $\mathbf{x}_j$ is the empty string. Moreover, for $h \in \mathbb{N}$ we denote by $\mathbf{x}_j^h$ the sub-string of $\mathbf{x}$ ranging from x_{j-h} to x_{j-1} if $j - h > 0$, and the prefix $\mathbf{x}_j$ otherwise. In other terms, $\mathbf{x}_j^h$ is the sub-string of $\mathbf{x}$ ranging from x_k to x_{j-1}, where k is the maximum between 1 and $j - h$. Notice that $\mathbf{x}_j^h$ has length either h or $j - 1$.

A language L is a set of strings over an alphabet Σ, i.e., $L \subseteq \Sigma^*$. Given a language L, we denote by $\overleftarrow{L}$ the mirror image of L, i.e., $\overleftarrow{L} = \{\overleftarrow{\mathbf{x}} \mid \mathbf{x} \in L\}$.

1.2 Quantum Computing

The most used model of Quantum Computation relies on the formalism of state vectors, unitary operators, and projectors. At high level we can say that state vectors evolve during the computation through unitary operators, then projectors remove part of the uncertainty on the internal state of the system.

The state of the system is represented by a unitary vector over the Hilbert space $\mathbb{C}^d$ with $d = 2^k$ for some $k \in \mathbb{N}$. The concept of bit of classical computation is replaced by that of qubit. While a bit can have value 0 or 1 a qubit is a unitary vector of $\mathbb{C}^2$. When the two components of the qubit are the complex numbers $\alpha = x + iy$ and $\beta = z + iw$, the squared norms $|\alpha|^2 = x^2 + y^2$ and $|\beta|^2 = z^2 + w^2$ represent the probabilities of measuring the qubit thus reading 0 and 1, respectively. In the more general case of k qubits the unitary vectors range in $\mathbb{C}^d$ with $d = 2^k$. Adopting the standard Dirac notation we denote a column vector $v \in \mathbb{C}^d$ by $|v\rangle$, and its conjugate transpose $v^\dagger$ by $\langle v|$. A *quantum state* is a unitary vector:

$$|\psi\rangle = \sum_{h=1}^{d} c_h |v_h\rangle$$

for some basis $\{|v_h\rangle\}$. In this case we also say that $|\psi\rangle$ is a *superposition* with coefficients $\{c_h\}$ over the basis $\{|v_h\rangle\}$. When not specified, we refer to the *canonical basis* denoted by $\{|0\rangle, |1\rangle, \ldots, |n-1\rangle\}$, where for each $q \in [0, d-1]$ the vector $|q\rangle$ is the unitary vector having 1 as $q+1$-th component and all its other components are 0. Moreover, usually $|q\rangle$ is written using the binary representation of q of length m. The canonical basis is an ortonormal basis for $\mathbb{C}^d$. Further details can be found in [28].

Unitary operators are a particular class of reversible linear operators. They preserve both the angles between vectors and their lengths. In other terms, unitary operators are transformation from one orthonormal basis to another. Hence, they are represented by unitary matrices. Let U be a square matrix over $\mathbb{C}$. U is said to be *unitary* iff $UU^\dagger = U^\dagger U = I$. We describe the application of a unitary matrix U to a state $|\psi\rangle$ by writing:

$$|\psi'\rangle = U |\psi\rangle$$

meaning that the state $|\psi\rangle$ becomes $|\psi'\rangle$ after applying the operator U.

In order to extract informations from a quantum state $|\phi\rangle$ a *measurement*, also called *observation*, must be performed. Projectors are the most common measurements/observables. Let $|u\rangle$ be a vector. The *projector* operator P_u along the direction of the unitary vector $|u\rangle$ is the linear operator defined as:

$$P_u = |u\rangle\langle u|$$

where $|u\rangle\langle u|$, being the product between a column vector and a row one both of size d, returns a matrix of size $d \times d$. Given a set of directions $F = \{|u_1\rangle, \ldots |u_f\rangle\}$ specified by unitary vectors the projector operator associated to F is defined as:

$$P_F = \sum_{u \in F} |u\rangle\langle u|$$

2 Measure-Once Quantum Finite Automata

Quantum Finite Automata (QFA) are the quantum counterpart of Finite Automata. Two models of Quantum Automata were independently introduced in the literature: Measure-Once QFAs (MO-QFAs) [27] and Measure-Many QFAs (MM-QFAs) [23]. The difference between the two definitions is about the number of observations that are made. While a MM-QFA is measured after reading each letter from the input, in a MO-QFA only one measurement is made after the whole input has been read.

We focus on MO-QFAs. Therefore, for sake of readability, we refer to MO-QFAs with just QFAs.

Let $\mathbb{C}^d$ be a finite dimension Hilbert space and $Q = \{|0\rangle, |1\rangle, \dots |d-1\rangle\}$ be its canonical basis. Usually in quantum computation it holds that $d = 2^k$ for some $k \in \mathbb{N}$, where k is the number of involved qubits. However, we refer here to a generic dimension d. It is not difficult to embed all the definitions and results we present into a space of dimension $2^{k'} > d$, thus using k' qubits, whenever it is necessary in the implementations.

Definition 1 (QFA). *A QFA is a* 5*-tuple* $M = (Q, \Sigma, \mathcal{U}, |\psi\rangle, F)$ *where:*

- Q*–the set of states– is the finite canonical basis of* $\mathbb{C}^d$ *for some* $d \in \mathbb{N}$*;*
- Σ *is a finite alphabet;*
- $\mathcal{U} = \{U_\sigma\}_{\sigma\in\Sigma}$ *is a finite set of unitaries of dimension* $\mathbb{C}^d \times \mathbb{C}^d$*;*
- $|\psi\rangle \in \mathbb{C}^d$ *is a unitary vector representing the initial superposition of* M*;*
- $F \subseteq Q$ *is the set of final states.*

In the literature the standard semantics attributed to QFA is based on the *Schrödinger picture* of quantum mechanics in which states evolve in time. We will come back to this in Section 3, when we will compare this interpretation with other possible ones. However, in the remaining of this section we will use the letter S of Schrödinger to refer to a generic QFA.

A generic configuration for a QFA S is a unitary vector of $\mathbb{C}^d$, i.e., it is a vector of the form:

$$|\varphi\rangle = \sum_{|q\rangle\in Q} \alpha_q |q\rangle$$

Let $|\varphi\rangle$ be the current configuration of S and $\sigma \in \Sigma$ be the current input symbol. $|\varphi\rangle$ evolves as follows:

$$|\varphi'\rangle = U_\sigma |\varphi\rangle$$

The computation starts from $|\psi\rangle$ and evolves reading the symbols of the string $\mathbf{x}$. At the end of the computation, i.e., when all the symbols of $\mathbf{x}$ have been read, a measurement is performed on the obtained state of S using the matrix $P_F = \sum_{|q\rangle\in F} |q\rangle\langle q|$. The probability of S accepting a string $\mathbf{x}$ is:

$$p_S(\mathbf{x}) = \|P_F U_{\mathbf{x}} |\psi\rangle\|^2 = \langle\psi| U_{\mathbf{x}}^\dagger P_F^\dagger P_F U_{\mathbf{x}} |\psi\rangle = \sum_{|q\rangle\in F} |\langle q| U_{\mathbf{x}} |\psi\rangle|^2$$

where $U_{\mathbf{x}}$–the evolution matrix accumulated along the read of $\mathbf{x}$– is defined as:

$$U_{\mathbf{x}} = U_{x_n} U_{x_{n-1}} \cdots U_{x_1}$$

We consider two different acceptance conditions. The first one is called with *cut-point* and it recalls the acceptance condition of probabilistic automata [32].

Definition 2 (Cut-point QFA). *A language $L \subseteq \Sigma^*$ is* accepted by a QFA S with cut-point λ *if and only if* $L = \{\mathbf{x} \in \Sigma^* \mid p_S(\mathbf{x}) > \lambda\}$.

A language $L \subseteq \Sigma^*$ is said to be accepted by a QFA with cut-point *if and only if there exist a QFA S and $\lambda \geq 0$ such that $L \subseteq \Sigma^*$ is accepted by S with* cut-point λ.

The second one is called with *certainty*. In this case we mimic the acceptance of a deterministic automata (DFA).

Definition 3 (Certainty QFA). *A language $L \subseteq \Sigma^*$ is said to be accepted by a QFA S with* certainty *if the following holds:*

$$\mathbf{x} \in L \textit{ iff } p_S(\mathbf{x}) = 1 \textit{ and } \mathbf{x} \notin L \textit{ iff } p_S(\mathbf{x}) = 0$$

It is straightforward to see that an acceptance with certainty implies an acceptance with cut-point $1 - \epsilon$, $\forall \epsilon \in (0, 1]$. The converse is trivially false.

The class of languages accepted by QFAs with *cut-point* was introduced and characterized in [14]. Such class is called *Unrestricted Measure-Once*, UMO. One of the main contribution to the characterization of such class is the connection with the languages accepted by Probabilistic Automata:

Theorem 1 ([14]). *Let L be a language accepted by a QFA S with cut-point λ. There exists a Probabilistic Finite Automaton that accepts L with cut-point λ', for some λ'.*

The class UMO was further investigated in [8,27], with the introduction of the following *pumping lemma*.

Theorem 2 ([27]). *Let $L \subseteq \Sigma^*$ be the language accepted by a QFA S with cut-point λ. $\forall \mathbf{x} = \mathbf{uv} \in L$ and $\forall \mathbf{y} \in \Sigma^*$, there exists $k \in \mathbb{N}^+$ such that $\mathbf{u}\mathbf{y}^k\mathbf{v} \in L$.*

A straightforward consequence of the above theorem is that finite languages cannot be accepted by QFA.

Corollary 1. *QFAs can accept only languages that are either empty or infinite.*

Notice that the theorem holds for any possible split of the string $\mathbf{x}$ into two strings $\mathbf{u}$ and $\mathbf{v}$. So, either $\mathbf{u}$ or $\mathbf{v}$ could be empty. In particular, taking $\mathbf{v}$ empty we get that languages whose elements have a fixed suffix cannot be recognized.

Corollary 2. *Let $\Sigma = \{a, b\}$ and $L = \{\mathbf{x} \,|\, \mathbf{x}$ ends with $a\}$. L cannot be accepted by any QFA S with cut-point.*

Proof. Suppose such a S exists. Let $\mathbf{x} \in L$ and $\mathbf{y} = b$, by Theorem 2 it holds that $\exists k \in \mathbb{N}^+$ such that $\mathbf{x}\mathbf{y}^k \in L$. This contradicts the definition of L. □

The above corollary also gives an example of a regular language that cannot be accepted by QFAs.

Despite being unable of accepting finite languages, QFAs can accept languages that are not regular. Let $\mathbf{x} \in \Sigma^*, \sigma \in \Sigma$. We denote by $|\mathbf{x}|_\sigma$ the number of occurrences of σ in $\mathbf{x}$. It was proven in [14] that there exists a QFA that accepts the language $L = \{\mathbf{x} \in \{a, b\}^* \ : \ |\mathbf{x}|_a \neq |\mathbf{x}|_b\}$ with cut-point 0.

The equivalent of UMO in the case of MM-QFAs is denoted by UMM (*Unrestricted Measure-Many*) and it was introduced in [14]. It was then characterized and eventually further investigated in the literature (see, e.g., [2]). Results on Quantum Automata descriptional complexity can be found in [12]. Recently in [18] the expressive power of Quantum Automata over the unary alphabet under different acceptance condition has been investigated. A physical realization of Quantum Automata has been presented in [26]. Undecidability results have been proved in [5]. In [11] it has been proved that languages accepted by MO-QFAs with bounded error are not definable in Linear Time Temporal Logics, while it is definable in the case of Measure-Many. A recent review can be found in [10].

Even more recently, Quantum Automata minimization has been studied in [22], while succintness has been described and implemented in [25]. Physical realizations of Quantum Computing algorithms always require to consider the noise introduced by non-perfect gates. In [13] the aim is to implement QFAs on noisy devices.

3 Heisenberg Quantum Finite Automata

The most widely adopted formulation of the Copenaghen interpretation of quantum mechanics is the *Schrödinger representation*. It is based on the idea that there is a *state vector* in an Hilbert space that completely describes the configuration of the system. This state vector evolves through time according to the *Schrödinger equation*. In particular, at each time instant a unitary operator is applied to the state vector. So, in the Schrödinger picture the state vector is time-dependent, while the unitaries and the observables remain unchanged.

There exists another representation known as *Heisenberg picture* in which the state vector is time-independent and always remains fixed to its value at time 0. Therefore, the time-dependency is *shifted* on the observables.

A third representation, named *Dirac picture*, also known as *Interaction picture*, "distributes" time dependencies over both states and operators.

Even though a mathematical equivalence between Schrödinger and Heisenberg representations has been proved by Von Neumann in [33], divergencies were pointed by Dirac in [16].

In terms of Quantum Finite Automata all the models described in the literature so far rely on the Schrödinger picture, where the initial state evolves through time using unitaries, while the observables never change [1].

In this section we shift to the Heisenberg picture and we formalize a new semantics for QFAs, named *Heisenberg Quantum Finite Automata* (HQFAs). The idea is that while the string $\mathbf{x}$ is read the state is unchanged, but there is an effect on the projector. At the end of the read such modified projector is applied to the initial state to obtain the final result. The way in which the observable gets modified is in a sense arbitrarily chosen. In our definition we try to keep such choice as close as possible to that of QFAs. In particular, in quantum mechanics when one shifts from the Schrödinger picture to the Heisenberg one a transformation of the states of the form $U|\varphi\rangle$ is mapped into a transformation of the observables/projectors of the form $U^\dagger PU$, where the meaning is that $U^\dagger$ has been applied to P. As a consequence HQFAs have exactly the same definition of QFAs, while the difference is in the acceptance condition, i.e., in the semantics.

Let P be the current observable of a HQFA H and $\sigma \in \Sigma$ be the current input symbol. P evolves as follows:

$$P' = U_\sigma^\dagger P U_\sigma$$

The computation starts from the observable P_F and evolves reading the symbols of $\mathbf{x}$. At the end of the read a measurement is performed using the resulting projector and the probability of accepting $\mathbf{x}$ is:

$$\rho_H(\mathbf{x}) = \| U_{\overleftarrow{\mathbf{x}}}^\dagger P_F U_{\overleftarrow{\mathbf{x}}} |\psi\rangle \|^2 = \langle\psi| U_{\overleftarrow{\mathbf{x}}}^\dagger P_F^\dagger P_F U_{\overleftarrow{\mathbf{x}}} |\psi\rangle$$

where consistently with the definition given in Section 2 the evolution matrix $U_{\overleftarrow{\mathbf{x}}}$ is defined as:

$$U_{\overleftarrow{\mathbf{x}}} = U_{x_1} U_{x_2} \cdots U_{x_n}$$

The acceptance condition with cut-point for an HQFA now inolves ρ_H.

Definition 4 (Cut-point HQFA). *A language $L \subseteq \Sigma^*$ is accepted by an HQFA H with cut-point λ if and only if $L = \{\mathbf{x} \in \Sigma^* \mid \rho_H(\mathbf{x}) > \lambda\}$.*

A language $L \subseteq \Sigma^*$ is said to be accepted by an HQFA with cut-point *if and only if there exist an HQFA H and $\lambda \geq 0$ such that $L \subseteq \Sigma^*$ is accepted by H with* cut-point λ.

Example 1. Let $Q = \{|0\rangle, |1\rangle\}$ be the canonical basis of $\mathbb{C}^2$. Let $\Sigma = \{a, b\}$. Consider the two unitary matrices $U_a = X$ (the negation gate) and $U_b = H$ (the Hadamard gate), i.e.:

$$U_a = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad U_b = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Let $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle$ and $F = \{|0\rangle\}$.

[1] Some work has been done for Quantum Cellular Automata, where the equivalence between Schrödinger model and Heisenberg model has been proved (e.g., [4]).

If we consider M as a QFA, i.e., we endow M with the the Schrödinger semantics, we get that the probability for the sting ab is:

$$p_M(ab) = \| |0\rangle \langle 0| U_b U_a |+\rangle \|^2 = \| |0\rangle \langle 0| U_b |+\rangle \|^2 = \| |0\rangle \langle 0|0\rangle \|^2 = \| |0\rangle \|^2 = 1$$

This means that no matter which is λ, the string ab is accepted.

If we consider the string abb we have to apply again U_b before projecting. Hence, we obtain $p_M(abb) = \| |0\rangle \langle 0| U_b |0\rangle \|^2 = \| |0\rangle \langle 0|+\rangle \|^2 = 1/2$.

On the other hand, if we look at M as a HQFA, i.e., we apply to M the Heisenberg semantics, the probability for the string ab is:

$$\rho_M(ab) = \|U_b^\dagger U_a^\dagger |0\rangle \langle 0| U_a U_b |+\rangle \|^2 = \|U_b^\dagger |1\rangle \langle 1| U_b |+\rangle \|^2 = \| |-\rangle \langle -|+\rangle \|^2 = 0$$

where $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. This means that no matter which is λ, the string ab is not accepted. Instead, if we consider the string ba we obtain:

$$\rho_M(ba) = \|U_a^\dagger U_b^\dagger |0\rangle \langle 0| U_b U_a |+\rangle \|^2 = \|U_a^\dagger |+\rangle \langle +| U_a |+\rangle \|^2 = \| |+\rangle \langle +|+\rangle \|^2 = 1$$

As a matter of fact, in this simple example one can notice that for all $\mathbf{x} \in \Sigma^*$ the behaviour of S on $\mathbf{x}$ is equivalent to the behaviour of H on its mirror image $\overleftarrow{\mathbf{x}}$, i.e., $p_S(\mathbf{x}) = \rho_H(\overleftarrow{\mathbf{x}})$. In the following we prove this result in the general case, for any automaton. □

Theorem 3. *Let M be a QFA over an alphabet Σ. For each $\mathbf{x} \in \Sigma^*$ it holds that*

$$p_M(\mathbf{x}) = \rho_M(\overleftarrow{\mathbf{x}})$$

Proof. Let $\mathbf{y} = \overleftarrow{\mathbf{x}}$. We have that $\overleftarrow{\mathbf{y}} = \mathbf{x}$. So, $\rho_M(\mathbf{y}) = \langle \psi | U_{\overleftarrow{\mathbf{y}}}^\dagger P_F^\dagger P_F U_{\overleftarrow{\mathbf{y}}} |\psi\rangle = \langle \psi | U_{\mathbf{x}}^\dagger P_F^\dagger P_F U_{\mathbf{x}} |\psi\rangle = p_M(\mathbf{x})$. □

Intuitively, when we shift to the Heisenberg picture, the effect of the first character of $\mathbf{x}$ is close to the observable instead of being close to the initial state. So, the word is read in the usual way from left to right by the automaton and the effects of the read are accumulated on the observable. However, when we look to such effects on the state, it is like if the word is read from right to left. In a sense it seems that the flow of time is reverted in the Heisenberg picture.

One could argue that we could have avoided the *mirror effect* by using in the Heisenberg definition the inverse unitary operators. Since the inverse of a unitary operator is its transposed conjugate, this would have meant to define the evolution of an observable P after reading a symbol σ as $P' = U_\sigma P U_\sigma^\dagger$. In the following example we show that such choice does not help in avoiding the mirroring.

Example 2. Let us consider again the automaton M defined in Example 1. The two matrices $U_a = X$ and $U_b = H$ coincide with their transposed conjugate, i.e., $U_a^\dagger = U_a$ and $U_b^\dagger = U_b$. So, the automaton M' defined using the transposed conjugate coincides with M. Hence, $p_M(\mathbf{x}) = \rho_{\mathbf{M'}}(\overleftarrow{\mathbf{x}})$, for any string $\mathbf{x}$. □

As a consequence of Theorem 3, the languages accepted by Heisenberg semantics are exactly the mirror images of those accepted by Schrödinger one.

Corollary 3. *Let $L \subseteq \Sigma^*$. L is accepted by a QFA with cut-point λ if and only if $\overleftarrow{L}$ is accepted by an HQFA with cut-point λ.*

So, now the question is whether the two formalisms have the same expressive power. As a consequence of the above corollary this is equivalent to check whether QFAs are closed under mirror images. By Example 1 we already know that it is not true that each language recognized by a QFA is closed under mirror images. However, it can be the case that whenever a language L is recognized by a QFA S, the language $\overleftarrow{L}$ is recongnized by a QFA S'.

Invoking Von Neumann's proof of equivalence of Schrödinger and Heisenberg pictures is not satisfactory by many point of views. First, Von Neumann's result has been proved in a general setting, while here we are confined in a restricted model, where there is a single initial state, while the final states are many. Moreover, only one projective measurement can be used and only at the end of the read. Second, we are not dealing with a single quantum system, but with an infinite set of systems, one for each string $\mathbf{x}$. The input $\mathbf{x}$ does not affect the initial state, but the sequence of unitary transformations. In a sense it affects the hamiltonian of the system. Third, it would be interesting to have either a constructive proof of equivalence or a counter-example in this specific setting.

The following result shows that QFAs are closed under mirror images. We provide a constructive proof. Given a QFA for a language L, we build a QFA for the language $\overleftarrow{L}$. Intuitively, the asymmetry between a single initial state and a set of final ones is solved through an opportune increase in the state space size.

Theorem 4 (Mirror Closure of QFAs). *Let $L \subseteq \Sigma^*$. L is accepted by a QFA with cut-point if and only if $\overleftarrow{L}$ is accepted by a QFA with cut-point.*

So, we can conclude that HQFAs do not increase QFAS expressive power.

Corollary 4 (Equivalence between QFAs and HQFAs). *L is accepted by a QFA with cut-point if and only if L is accepted by an HQFA with cut-point.*

Proof. Let L be accepted by a QFA with cut-point. By Theorem 4 $\overleftarrow{L}$ is accepted by a QFA with cut-point. As a consequence of Corollary 3 L is accepted by an HQFA with cut-point.

On the other hand, let L be accepted by an HQFA with cut-point. By Corollary 3 $\overleftarrow{L}$ is accepted by a QFA with cut-point. By Theorem 4 L is accepted by a QFA with cut-point. □

4 Heisenberg inspired Automata: (Un)bounded Memory

The Heisenberg semantics introduced in the previous section has the same expressive power of the Schrödinger one introduced in the literature. However, we can take inspiration from Heisenberg proposal and analyse what happens if each time a character is read all the unitary matrices are transformed, i.e., instead of changing at each step the observables we modify the unitaries associated to the single characters. We do such changes by exploiting the characters that have already been read.

In particular, given an automaton $M = (Q, \Sigma, \{U_\sigma\}_{\sigma\in\Sigma}, |\psi\rangle, F)$, after reading the prefix $\mathbf{x}_j$ of the string $\mathbf{x} = x_1x_2\dots x_n$ the unitary matrix associated to a character σ has evolved into:

$$\mathbb{W}_\sigma^{\mathbf{x}_j} = U_{\mathbf{x}_j}U_\sigma$$

where $U_\epsilon = Id$ is the identity transformation. So, if the current configuration after reading $\mathbf{x}_j$ is $|\varphi\rangle$ and we read x_j, then the state evolves as follows:

$$|\varphi'\rangle = \mathbb{W}_{x_j}^{\mathbf{x}_j}|\varphi\rangle = U_{\mathbf{x}_j}U_{x_j}|\varphi\rangle$$

The computation starts from $|\psi\rangle$ and evolves reading the symbols of the string $\mathbf{x}$. The state reached at the end of the read is:

$$\mathbb{W}_{\mathbf{x}}|\psi\rangle$$

where $\mathbb{W}_{\mathbf{x}}$–the evolution matrix accumulated along the read of $\mathbf{x}$– is defined as:

$$\mathbb{W}_{\mathbf{x}} = \mathbb{W}_{x_n}^{\mathbf{x}_n}\mathbb{W}_{x_{n-1}}^{\mathbf{x}_{n-1}}\cdots\mathbb{W}_{x_2}^{\mathbf{x}_2}\mathbb{W}_{x_1}^{\mathbf{x}_1}$$

As in the case of QFAs the projector P_F is finally applied to obtain the probability of accepting a string $\mathbf{x}$, denoted by $\omega_M(\mathbf{x})$:

$$\omega_M(\mathbf{x}) = \|P_F\mathbb{W}_{\mathbf{x}}|\psi\rangle\|^2 = \langle\psi|\mathbb{W}_{\mathbf{x}}^\dagger P_F^\dagger P_F\mathbb{W}_{\mathbf{x}}|\psi\rangle$$

Example 3. Let us consider again the automaton of Example 1. If we consider the string *abb* the evolution matrix that is applied to the initial state is:

$$[(U_bU_a)U_b][(U_a)U_b][(Id)U_a]$$

where we use the parenthesis to emphasized the single steps. In particular, the squared parenthesis enclose the read of a single character, while the rounded ones enclose the transformations due to the read of the prefix accumulated so far. Instantiating U_a, U_b and $|\psi\rangle$ as in Example 1 we obtain that the state reached at the end of the read is $-|1\rangle$. So, since $F = \{|0\rangle\}$, we get:

$$\omega_M(abb) = (-\langle 1|)P_F^\dagger P_F(-|1\rangle) = 0$$

Example 4. Let us now consider a simpler example in which $\Sigma = \{a\}$, $U_a = X$, the initial state is $|\psi\rangle = |0\rangle$, and $F = \{|0\rangle\}$. It is immediate to see that when a string of the form a^k is read the evolution matrix has the form:

$$X^kX^{k-1}\cdots X^2X = X^{\frac{(k+1)k}{2}}$$

This means that a string of length k is accepted by the automaton if and only if $(k+1)k$ is a multiple of 4. □

As in the case of HQFAs, for these automata, that we call UMQFAs (*Unbounded Memory Quantum Finite Automata*), the syntactic definition is the same as for QFAs, while the accepting condition is different.

Definition 5 (Cut-point UMQFA). *A language $L \subseteq \Sigma^*$ is accepted by a UMQFA M with cut-point λ if and only if $L = \{\mathbf{x} \in \Sigma^* \mid \omega_M(\mathbf{x}) > \lambda\}$.*

A language $L \subseteq \Sigma^*$ is said to be accepted by a UMQFA with cut-point *if and only if there exist a UMQFA M and $\lambda \geq 0$ such that $L \subseteq \Sigma^*$ is accepted by M with* cut-point λ.

Notice that we arbitrarily decided to rely on a single set of matrices. One could have considered a more general definition. The only important point is that when a character is read the unitary matrix that is applied depends also on all the characters that have been read before. However, such dependency have to be defined in a finitary way, i.e., relying on a finite initial set of matrices.

So the question now becomes: is this semantics increasing the expressive power of QFAs? In order to analyse such question we first take a step back and study what happens when, instead of using all the characters that have been read so far, we only use a bounded amount of them. On the one hand, such step back makes the situation more similar to what happen in the case of classical automata, which have a finite amount of memory. On the other hand, this naturally allows to give a more general definition, where a larger set of unitaries is used.

4.1 Bounded Memory

The most natural way to instantiate the above semantics in order to take care only of a bounded quantity of characters is to fix $h \geq 0$ and to refer to $\mathbf{x}_j^h$ instead of $\mathbf{x}_j$ (see Section 1.1). The sub-string $\mathbf{x}_j^h$ takes into account at most h symbols that precede x_j in the string $\mathbf{x}$. Since there exists a finite number of strings of length at most h, the matrices $W_\sigma^{\mathbf{y}}$, with $\mathbf{y}$ of length at most h, can be directly specified in the definition of the automaton. Such automata have been already defined in the literature [30,6], using an equivalent notation, and called *Multi-letter Quantum Finite Automata (MQFA)*. However, as we will discuss a different acceptance condition was used. Let $h \in \mathbb{N}$.

Definition 6 (h-MQFA). *An h-MQFA is a 5-tuple $M = (Q, \Sigma, \mathcal{W}, |\psi\rangle, F)$ where:*

- *Q–the set of states– is the finite canonical basis of $\mathbb{C}^d$ for some $d \in \mathbb{N}$;*
- *Σ is a finite alphabet;*
- *$\mathcal{W} = \{W_\sigma^{\mathbf{y}}\}_{\sigma \in \Sigma, \mathbf{y} \in \Sigma^{\leq h}}$ is a finite set of unitaries of dimension $\mathbb{C}^d \times \mathbb{C}^d$;*
- *$|\psi\rangle \in \mathbb{C}^d$ is a unitary vector representing the initial superposition of M;*
- *$F \subseteq Q$ is the set of final states.*

Let $\mathbf{x} = x_1 x_2 \dots x_n$ be an input string for an h-MQFA $M = (Q, \Sigma, \mathcal{W}, |\psi\rangle, F)$. The computation starts in the state $|\psi\rangle$. Let us assume that after reading the first $j-1$ symbols of $\mathbf{x}$ a state $|\varphi\rangle$ is reached. When x_j is read the states evolves according to the following law:

$$|\varphi'\rangle = W_{x_j}^{\mathbf{x}_j^h} |\varphi\rangle$$

The computation starts from $|\psi\rangle$ and evolves reading the symbols of the string $\mathbf{x}$. At the end of the computation, a measurement is performed on the state of M through the projector P_F. The probability of M accepting $\mathbf{x}$ is:

$$\mu_M(\mathbf{x}) = \| P_F W_{\mathbf{x}} |\psi\rangle \|^2$$

where $W_{\mathbf{x}}$ is defined as:

$$W_{\mathbf{x}} = W_{x_n}^{\mathbf{x}_n^h} W_{x_{n-1}}^{\mathbf{x}_{n-1}^h} \cdots W_{x_2}^{\mathbf{x}_2^h} W_{x_1}^{\mathbf{x}_1^h}$$

The acceptance condition with cut-point for an h-MQFA is based on μ_M.

Definition 7 (Cut-point h-MQFA). *A language $L \subseteq \Sigma^*$ is accepted by an h-MQFA M with cut-point λ if and only if $L = \{\mathbf{x} \in \Sigma^* \mid \mu_M(\mathbf{x}) > \lambda\}$.*

A language $L \subseteq \Sigma^*$ is said to be accepted by an h-MQFA with cut-point *if and only if there exist an h-MQFA M and $\lambda \geq 0$ such that $L \subseteq \Sigma^*$ is accepted by M with* cut-point λ.

Intuitively, h-MQFAs have bounded memory h in the sense that at each point of the computation the preceding h characters are used for choosing the evolution. Notice that QFAs coincide with 0-MQFAs. Moreover, each h'-MQFA can be embedded into a h-MQFA with $h > h'$ by simply defining $\mathcal{W}$ in such a way that if $\mathbf{x}$ and $\mathbf{y}$ are two strings of length at most h that coincide on the suffix of length h', then $W_\sigma^{\mathbf{x}} = W_\sigma^{\mathbf{y}}$, for each $\sigma \in \Sigma$.

Example 5. Let $\Sigma = \{a, b\}$ and $L = \{a, b\}^* b$, i.e., the language of strings that end with b. The pumping lemma for QFAs ensures that this language cannot be accepted by a QFA with cut-point. Consider instead the 1-MQFA $M = (Q, \Sigma, \mathcal{W}, |\psi\rangle, F)$ where $Q = \{|0\rangle, |1\rangle\}$, $\Sigma = \{a, b\}$, $|\psi\rangle = |0\rangle$, $F = \{|1\rangle\}$. The set $\mathcal{W}$ is defined as follows:

$$W_b^\epsilon = W_b^a = W_a^b = X \qquad W_b^b = Id$$

and all the other matrices are the identity. The above matrices exactly simulate the behaviour of the following deterministic automaton, interpreting $|i\rangle$ as q_i:

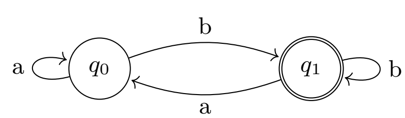

Fig. 1. Deterministic automaton accepting $\{a, b\}^* b$.

So, M accepts L with certainty, hence also any cut-point $\lambda \geq 0$ is fine.

In [6,30] properties of this class of automata have been studied in the case of isolated cut-point acceptance condition, also called bounded error. It was shown that the expressive power of h-MQFAs is strictly dependent on the parameter h. The set of languages recognized by h-MQFAs with bounded error coincides with those recognized by h-Group Finite Automata and are a subset of regular languages. As a consequence in [30] it has been proved that the set of languages accepted by a h'-MQFAs with bounded error is strictly included in the set of languages accepted by h-MQFAs with bounded error, for $h' < h$. This is consistent with our intuition that more memory increases the computation power. Moreover, in [30], it was proved that if the minimal DFA accepting a language L contains a particular forbidden structure, then L cannot be accepted by h-MQFAs with bounded error, for any $h \geq 0$. This is a structural characterization of languages that cannot be accepted by h-MQFAs with bounded error.

In this section, as in the rest of this paper, we focus on cut-point acceptance condition which is less demanding than bounded error and has not been studied in the literature for h-MQFAs. We start presenting a *pumping lemma* which provides a structural characterization of the languages that are accepted from h-MQFAs with cut-point. Then we investigate on the expressive power of h-MQFAs with respect to h. Differently from QFAs, h-MQFAs can also recognize finite languages and still they constitute a proper hierarchy.

In the proof of the pumping lemma we exploit the following lemma which is also at the basis of the pumping lemma for QFAs. The norm $\|A\|$ of a matrix A is defined as:

$$\|A\| = \sup_{\langle u|u\rangle=1} \{\|A\,|u\rangle\,\|\}$$

Lemma 1 ([8]). *Let $V \in \mathbb{C}^d \times \mathbb{C}^d$ be a unitary matrix let $Id \in \mathbb{C}^d \times \mathbb{C}^d$ be the identity matrix of dimension d. For any $\varepsilon > 0$ there exists $k \in N^+$ such that:*

$$\|Id - V^k\| \leq \varepsilon$$

The pumping lemma for h-MQFAs states that if we consider a sufficiently long suffix of a string which is inside the accepted language, then we can pump such suffix for an opportune number of times and fall again inside the language.

Theorem 5 (Pumping Lemma for h-MQFAs). *Let $L \subseteq \Sigma^*$ be the language accepted by an h-MQFA. Then, $\forall \mathbf{x} = \mathbf{uv} \in L$ with $|\mathbf{v}| \geq h$ there exists $k \in \mathbb{N}^+$ such that $\mathbf{xv}^k \in L$.*

Notice that differently from Theorem 2, the above pumping lemma does not prevent finite languages to be accepted by h-MQFAs. As a matter of fact, if all the strings accepted by an h-MQFAs are shorter than h, then it is not possible to find a suffix that can be pumped.

Theorem 6 (Singleton/Finite Languages). *Let $L = \{\mathbf{w}\}$ with $\mathbf{w} \in \Sigma^{h-1}$ and $h - 1 > 0$. Then there exists an h-MQFA that accepts L with certainty.*

Let L be a finite language whose elements have length less than h. There exists an h-MQFA that accepts L with cut-point.

We can exploit our pumping lemma to prove that the amount of memory we provided to the h-MQFA in the above theorem is the minimum.

Lemma 2. *Let $L = \{\mathbf{w}\}$ with $\mathbf{w} \in \Sigma^{h-1}, h-1 > 0$. Then, there is no h'-MQFA, with $h' < h$ that accepts L with cut-point.*

Proof. Assume by contradiction that there exists an h'-MQFA that accepts L with $h' < h$. Since $|\mathbf{w}| = h - 1 \geq h'$, the string $\mathbf{w}$ can be written as $\mathbf{uv}$ with $\mathbf{v} \in \Sigma^{h'}$. By Theorem 5, it holds that there exists a $k \in \mathbb{N}^+$ such that $\mathbf{wv}^k \in L$. So, we have a contradiction. □

Exploiting Theorems 5 and 6, together with Lemma 2 we have that the set of languages accepted by h'-MQFAs is a proper subset of the set of languages accepted by h-MQFAs, with $h' < h$. The inclusion immediately follows from the definition of h-MQFAs and our results show that the inclusion is proper by exhibiting as witnesses all the singleton languages of strings of length h'.

Corollary 5. *The set of languages accepted by h'-MQFAs with cut-point is a proper subset of those accepted by h-MQFAs with cut-point, when $h' < h$.*

The hierarchy result proved in [30] concerns sets of languages which are all included in the set of regular languages, while our hierarchy includes already at level 0 non-regular languages.

4.2 Unbounded Memory

QFAs and also h-MQFAs fail to recognize many classical regular languages, since unitary transformations introduce a notion of memory which is quite different from the classical one.

On the one hand, it is easy to define a classical automaton for a finite language by using the finite set of states of the automaton to store the finite quantity of memory that is necessary. This cannot be achieved in QFAs and in h-MQFAs, when h is not large enough, as a consequence of the following property of unitary matrices that has been stated in Lemma 1:

$$\forall \varepsilon > 0 \, \exists k \in \mathbb{N}^+ \ \ \|Id - V^k\| \leq \varepsilon$$

This is the key ingredient of the pumping lemmas for QFAs and h-MQFAs.

On the other hand, it is possible to define a QFA that accepts the non-regular language of strings having a different number of a and b characters. Classical automata do not have enough memory for this language, since it is necessary to count an unbounded number of characters.

We started this section introducing the Heisenberg inspired automata called UMQFAs hoping to increase the expressive power of QFA and h-MQFAs still relying on a finite set of unitaries and a single measurement at the end of the read. It is time to draw some conclusions about this. The automaton described in Example 4 pointed out that in UMQFAs we are not able to replicate the use of a unitary matrix V for any possible $k \in \mathbb{N}^+$, i.e., we cannot exploit Lemma 1. For instance in the example the matrix X can only occur with an exponent of the form $(k+1)k/2$, i.e., all the possible values assumed by a polynomial $p(k)$ when k ranges in $\mathbb{N}^+$. The proof of Lemma 1 in [8] is based on Cauchy sequences and cannot be easily generalized. However, there is another proof of the same result in [14] that ultimately relies on the following algebraic property:

for each $\alpha \in \mathbb{R} \setminus \mathbb{Q}$ the set of fractional parts of the multiples of α, i.e., $\{k\alpha - \lfloor k\alpha \rfloor \mid k \in \mathbb{N}\}$, is dense in $[0, 1]$.

This results generalizes to polynomials having irrational coefficients and to multiple dimensions (e.g., [7]). As a consequence we have a language that can be accepted by h-MQFAs, but not by UMQFAs.

Theorem 7. *Let $\Sigma = \{a\}$ and $L = \{\epsilon, a\}$. There is a 2-MQFA that accepts L with cut-point and there is not a UMQFAs that accepts L with cut-point.*

There are technical ingredients in the proof of the above result that are somehow interesting. We had to carefully choose the language L in order to obtain homogeneous polynomials. Otherwise the eigenvalues related to rotations that are rational multiples of π would have given troubles. Moreover, the interplay between some eigenvalues could be favorable for constructing UMQFAs that *approximate* h-MQFAs, since the distribution of the wrong strings accepted by the UMQFA is not uniform.

Beside these technical considerations, the result shows that the unbounded memory we tried to introduce does not generalize the bounded one, and it does not seem easy to find a natural generalization with a finitary description.

5 Conclusions

Quantum Computing is becoming a more and more investigated subject thanks to phenomena like quantum speed-up. Using the properties of quantum mechanics it is possible to design algorithms that polynomially solve problems that require exponential time with classical computation [28]. However, when one looks at basic models of computation such as automata the rules of quantum mechanics, imposing unitary evolutions along the computation, constitute more an obstacle to the expressive power, than an advantage. Informally, we can say that the unitaries cause a loss of memory in the automata. As a matter of fact, a simple language including only one string cannot be accepted by MO-QFAs.

In our work we tried to better understand the role of unitaries and measurements in MO-QFAs. We proved that for any MO-QFA there is a "reversed" MO-QFA that accepts the mirror language. Then we analysed the effect of playing with the unitaries. We forced a sort of stuttering behaviour hoping to gain expressive power. We obtained a first negative result which however gives some suggestions for further investigations. For example, there may be other definitions for the Unbounded Memory case that lead to larger expressive power.

Appendix: Proofs of Main Theorems

5.1 Proof of Theorem 4

Let $M = (Q, \Sigma, \{U_\sigma\}_{\sigma\in\Sigma}, |\psi\rangle, F)$ be a QFA accepting L with cut-point λ. We recall that Q is the canonical basis of $\mathbb{C}^d$, for some $d \in \mathbb{N}$. Without loss of generality, let $F = \{q_0, q_1, \dots q_{m-1}\}$. Let $\mathbf{x} = x_1 x_2 \dots x_n$ be an input string. By definition, the acceptance probability of M for $\mathbf{x}$ is:

$$p_M(\mathbf{x}) = \sum_{i=0}^{m-1} |\langle q_i| U_{\mathbf{x}} |\psi\rangle|^2$$

We now define $\overleftarrow{M} = (\overleftarrow{Q}, \Sigma, \{V_\sigma\}_{\sigma\in\Sigma}, \left|\overleftarrow{\psi}\right\rangle, \overleftarrow{F})$, where $\overleftarrow{Q}$ is the canonical basis of $\mathbb{C}^{dm}$ and:

$$V_\sigma = \sum_{i=0}^{m-1} |i\rangle\langle i| \otimes U_\sigma^\dagger, \quad \left|\overleftarrow{\psi}\right\rangle = \frac{1}{\sqrt{m}} \sum_{i=0}^{m-1} |i\rangle \otimes |q_i\rangle, \quad \overleftarrow{F} = \{|i\rangle \otimes |\psi\rangle \mid i \in [0, m-1]\}$$

We have that $V_{\mathbf{x}} = V_{x_n} V_{x_{n-1}} \cdots V_{x_1}$ and $U_{\overleftarrow{\mathbf{x}}} = U_{x_1} U_{x_2} \dots U_{x_n}$. By definition of QFA we get:

$$\begin{aligned}
p_{\overleftarrow{M}}(\mathbf{x}) &= \|P_{\overleftarrow{F}} V_{\mathbf{x}} \left|\overleftarrow{\psi}\right\rangle\|^2 \\
&= \|P_{\overleftarrow{F}} \frac{1}{\sqrt{m}} \sum_{i=0}^{m-1} |i\rangle \otimes U_{x_n}^\dagger U_{x_{n-1}}^\dagger \cdots U_{x_1}^\dagger |q_i\rangle\|^2 \\
&= \|\frac{1}{\sqrt{m}} \sum_{i=0}^{m-1} |i\rangle \otimes \Big(|\psi\rangle\langle\psi| U_{x_n}^\dagger U_{x_{n-1}}^\dagger \cdots U_{x_1}^\dagger |q_i\rangle\Big)\|^2 \\
&= \|\frac{1}{\sqrt{m}} \sum_{i=0}^{m-1} \Big(\langle\psi| U_{x_n}^\dagger U_{x_{n-1}}^\dagger \cdots U_{x_1}^\dagger |q_i\rangle\Big) |i\rangle \otimes |\psi\rangle\|^2 \\
&= \frac{1}{m} \sum_{i=0}^{m-1} |\langle\psi| U_{x_n}^\dagger U_{x_{n-1}}^\dagger \cdots U_{x_1}^\dagger |q_i\rangle|^2 \\
&= \frac{1}{m} \sum_{i=0}^{m-1} |(\langle q_i| U_{\overleftarrow{\mathbf{x}}} |\psi\rangle)^*|^2 = \frac{1}{m} \sum_{i=0}^{m-1} |\langle q_i| U_{\overleftarrow{\mathbf{x}}} |\psi\rangle|^2 = \frac{1}{m} p_M(\overleftarrow{\mathbf{x}})
\end{aligned}$$

Let $\overleftarrow{\lambda} = \frac{\lambda}{m}$. We have that:

$$\overleftarrow{\mathbf{x}} \in L \text{ iff } p_M(\overleftarrow{\mathbf{x}}) > \lambda \text{ iff } p_{\overleftarrow{M}}(\mathbf{x}) > \overleftarrow{\lambda} \text{ iff } \mathbf{x} \in \overleftarrow{L}$$

□

5.2 Proof of Theorem 5

For sake of readability we prove the result for $|\mathbf{v}| = h$. For $|\mathbf{v}| > h$ the idea is the same, just the notation would be much heavier.

Let $L \subseteq \Sigma^*$ be a language and let $M = (Q, \Sigma, \mathcal{W}, |\psi\rangle, F)$ be an h-MQFA that accepts L with *cut-point* λ.

Let $\mathbf{x} = \mathbf{uv} = u_1 \dots u_a v_1 \dots v_h$ be a string of L. First we write the matrix $W_{\mathbf{xv}^j}$, for a generic $j \in \mathbb{N}$, in order to make explicit its relationship with $W_{\mathbf{x}}$. In particular, by applying the definition of h-QMFA we have that $W_{\mathbf{xv}^j} = V^j W_{\mathbf{x}}$, where V is defined as:

$$V = W_{v_h}^{v_h v_1 \cdots v_{h-1}} W_{v_{h-1}}^{v_{h-1} v_h v_1 \cdots v_{h-2}} \dots W_{v_1}^{v_1 v_2 \cdots v_h}$$

Having represented a vector $|v\rangle$ in the canonical basis and being $|q\rangle$ be an element of the canonical basis, let $(|v\rangle)_q$ be the $q-$th component of $|v\rangle$. For any $j \in \mathbb{N}$ it holds:

$$\begin{aligned}
&|\mu_M(\mathbf{x}) - \mu_M(\mathbf{xv}^j)| = \left|\sum_{q\in F} \left(|(W_{\mathbf{x}} |\psi\rangle)_q|^2 - |(W_{\mathbf{xv}^j} |\psi\rangle)_q|^2\right)\right| \leq \\
&\leq 2\sum_{q\in F} |\,|(W_{\mathbf{x}} |\psi\rangle)_q| - |(W_{\mathbf{xv}^j} |\psi\rangle)_q|\,| \leq 2\sum_{q\in F} |(W_{\mathbf{x}} |\psi\rangle)_q - (W_{\mathbf{xv}^j} |\psi\rangle)_q| = \\
&= 2\sum_{q\in F} \left|\, (W_{\mathbf{x}} |\psi\rangle)_q - (V^j W_{\mathbf{x}} |\psi\rangle)_q \,\right| = 2\sum_{q\in F} \left|\, \langle q| \,(Id - V^j) W_{\mathbf{x}} |\psi\rangle \,\right| \leq \\
&\leq 2\sum_{q\in F} \|Id - V^j\| = 2|F|\|Id - V^j\|
\end{aligned}$$

Since $\mathbf{x} \in L$, we have $\mu_M(x) - \lambda = \Delta > 0$. By Lemma 1 there exists $k \in \mathbb{N}^+$ such that:

$$\|Id - V^k\| \leq \frac{\Delta}{4|F|}$$

which yields to $\left|\mu_M(\mathbf{x}) - \mu_M(\mathbf{xv}^k)\right| \leq \frac{\Delta}{2}$. Therefore, $\mathbf{xv}^k \in L$, since:

$$\mu_M(\mathbf{xv}^k) - \lambda \geq \mu_M(\mathbf{x}) - \frac{\Delta}{2} - \lambda \geq \frac{\Delta}{2} \geq 0$$

□

5.3 Proof of Theorem 6

Let $M = (Q, \Sigma, \mathcal{W}, |\psi\rangle, F)$ be a h-MQFA, where $Q = \{|0\rangle, |1\rangle\}$, $|\psi\rangle = |0\rangle$, $F = \{|1\rangle\}$. The states $|0\rangle$ and $|1\rangle$ are such that $|1\rangle = X|0\rangle$ and $|0\rangle = X|1\rangle$. Since $\mathbf{w}$ has length $h - 1 > 0$, $\mathbf{w} = \mathbf{u}\alpha$ with $\mathbf{u} \in \Sigma^{h-2}, \alpha \in \Sigma$. We define:

$$\begin{aligned}
&W_\alpha^{\mathbf{u}} = X \\
&W_\sigma^{\mathbf{w}} = X \quad \forall \sigma \in \Sigma
\end{aligned}$$

while all the other matrices inside $\mathcal{W}$ are the identity matrix. We must now prove that the language accepted by M is exactly L.

If $\mathbf{w}$ is the input for M, then the computation evolves as follows:

$$W_{\mathbf{w}} = W_m^{\mathbf{w}_m^h} W_{w_{m-1}}^{\mathbf{w}_{m-1}^h} \dots W_{w_1}^{\mathbf{w}_1^h}$$

Since all the matrices we set to be different from the identity concern strings with length that is at least $h-1$, it holds that $W_{w_j}^{\mathbf{w}_j^{h+1}} = I, \forall j \in \{1, 2, \ldots m-1\}$ Therefore,

$$W_{\mathbf{w}} = W_{w_m}^{\mathbf{w}_m^{h+1}} = W_{\alpha}^{\mathbf{u}} = X$$

Since the initial state is $|0\rangle$, then $\|PW_{\mathbf{w}}|0\rangle\|^2 = \|P|1\rangle\|^2 = 1$.

Otherwise, suppose $\mathbf{x} \neq \mathbf{w}$ is the input for M. If the string $\mathbf{x}$ does not contain the sub-string $\mathbf{w}$, then clearly $W_{\mathbf{x}} = Id$, and $\mathbf{x}$ is refused. If $\mathbf{x}$ has $\mathbf{w}$ as proper prefix, then $\mathbf{x}$ is of the form $\mathbf{ws}$, with $\mathbf{s} = \sigma_1 \ldots \sigma_j$, $j \geq 1$. In this case, we have that $W_{\mathbf{x}}$ is as follows:

$$\begin{aligned} W_{\mathbf{x}} &= W_{\sigma_j}^{\mathbf{x}_{h-j-1}^h} W_{\sigma_{j-1}}^{\mathbf{x}_{h-j-2}^h} \cdots W_{\sigma_1}^{\mathbf{x}_{h-1}^h} W_{\mathbf{w}} \\ &= W_{\sigma_j}^{\mathbf{x}_{h-j-1}^h} W_{\sigma_{j-1}}^{\mathbf{x}_{h-j-2}^h} \cdots W_{\sigma_1}^{\mathbf{w}} W_{\mathbf{w}} = Id \cdots XX = Id \end{aligned}$$

since all the matrices of the form $W_{\sigma}^{\mathbf{y}}$, with $\mathbf{y}$ of length h are the identity matrix. So, $\mathbf{x}$ is refused. The last case we need to consider is when $\mathbf{w}$ occurs as a proper sub-string of $\mathbf{x}$, but it is not a proper prefix of $\mathbf{x}$. This means that the input $\mathbf{x}$ is of the form $\mathbf{x} = \mathbf{vws}$, with $\mathbf{v} \neq \epsilon$ and $\mathbf{vw}$ which does not have $\mathbf{w}$ as prefix. In this case, the key point is that since $|\mathbf{w}| = h-1$, but $\mathbf{w}$ is now preceded by at least one character the matrices $W_{\alpha}^{\mathbf{u}}$ and $W_{\sigma}^{\mathbf{w}}$ do not occur in $W_{\mathbf{x}}$. So, $W_{\mathbf{x}} = Id$ and $\mathbf{x}$ is refused. Notice that the automaton we defined accepts with certainty.

Let $L = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\}$ be a finite language whose elements have length less than h. For each element $\mathbf{x}_j$ there exists an h_j-MQFA that accepts only $\mathbf{x}_j$ with certainty. As already observed any h'-MQFA can be embedded into an h-MQFA that accepts the same language with the same cut-point, if $h \geq h'$. Let h be greater than the length of the longest string in L. We have that for each element $\mathbf{x}_j$ of L there exists an h-MQFA M_j that accepts only $\mathbf{x}_j$ with certainty. The tensor product M of the M_j's automata, whose construction is similar to that used in the proof of Theorem 4 accepts the language L. The tensor product automaton does not accept with certainty but with cut-point λ, with λ any number in the interval $(0, 1/\ell)$. □

5.4 Proof of Corollary 5

Let $h, h' \in \mathbb{N}^+$ with $h' < h$. From Lemma 2 we know that there exist languages accepted by $h-$MQFAs, but not by $h'-$MQFAs.
We must prove that all the languages accepted by $h'-$MQFAs are also accepted by $h-$MQFAs.
Let $M = (Q, \Sigma, \mathcal{W}, |\psi\rangle, F)$ be a $h'-$MQFA accepting a language L. We can build an $h-$MQFA $M' = (Q, \Sigma, \mathcal{W}', |\psi\rangle, F)$ accepting the same language setting $\mathcal{W}' = \mathcal{W}$ (eventually completing with identity matrices). □

5.5 Proof of Theorem 7

By Theorem 6 there is a 2-MQFA that accepts $L = \{\epsilon, a\}$ with cut-point.

Let us assume by contradiction that there exists $M = (Q, \Sigma, \mathcal{U}, |\psi\rangle, F)$ UMQFA that accepts $L = \{\epsilon, a\}$ with cut-point λ. Since the string ϵ is in L it has to be:

$$\omega_M(\epsilon) = \| P_F |\psi\rangle \|^2 = \| P_F U_a |\psi\rangle \|^2 = \lambda + \Delta > \lambda$$

Any other string a^k, with $k > 1$ over the alphabet Σ would instead give:

$$\omega_M(a^k) = \| P_F \mathbb{W}_{a^k} |\psi\rangle \|^2 = \| P_F U_a^{\frac{k(k+1)}{2}} |\psi\rangle \|^2$$

Let us consider a generic unitary matrix V and study the sequence:

$$\{V^{\frac{k(k+1)}{2}}\}_{k>1}$$

As observed in [14], V can be diagonalized and $V^h = RD^hR^{-1}$, where R is unitary and D is the diagonal matrix of the eigenvalues of V. Let $e^{i\pi v_j}$ be the j-th eigenvalue of V.

If all the r_js are rational, then let $n = 4\Pi_j q_j$, where the q_js are the denominators of the r_js. We have that $D^{\frac{n(n+1)}{2}} = Id$, and hence $V^{\frac{n(n+1)}{2}} = Id$.

If m of the r_j are irrational, and ℓ of them are rational, we can safely assume that the first m are the irrational ones. Let again n be defined as above considering only the rational coefficients. If we consider the sub-sequence:

$$\{V^{\frac{nk(nk+1)}{2}}\}_{k>1}$$

we have that all the rational eigenvalues have always values 1 in the sub-sequence. On the other hand, the remaining eigenvalues take values of the form $e^{i\pi r_j \frac{nk(nk+1)}{2}}$ in the sub-sequence. Let $p : \mathbb{N} \to \mathbb{R}^m$ be defined as $p(k) = (r_1 4nk(4nk+1), \ldots, r_m 4nk(4nk+1))$. These are quadratic polynomials in the variable k with irrational coefficients. The fractional parts of each of these polynomials are dense and uniformly distributed over $[0, 1]$ (e.g., [7]). This means that each of these polynomials is infinitely many times arbitrarily close to a multiple of 4. This implies that each of the values $e^{i\pi r_j \frac{nk(nk+1)}{2}}$ is infinitely many times arbitrarily close to 1. As far as the whole polynomial function p is concerned it is uniformly distributed over $[0, 1]^m$ if the irrational r_js are independent. When some of the of the irrational r_js are linear combinations of the others the uniform distribution is no more ensured, but the density in $(0, 0, \ldots, 0)$ is preserved, since by making the fractional parts of the independent ones arbitrary small we can ensure that also the fractional parts of their linear combination are small enough.

Hence, for any unitary matrix V, and for each ε there exists $k > 1$ such that:

$$\| Id - V^{\frac{k(k-1)}{2}} \| \leq \varepsilon$$

As a consequence working as in the proof of Theorem 5 on the string ϵ which is in L and using $U_a^{\frac{k(k-1)}{2}}$ we obtain that there exist $k > 1$ such that a^k is accepted by M. This is a contradiction. □

References

1. Ambainis, A., Beaudry, M., Golovkins, M., Kikusts, A., Mercer, M., Thérien, D.: Algebraic results on quantum automata. Theory of Computing Systems **39**(1), 165–188 (2006)
2. Ambainis, A., Freivalds, R.: 1-way quantum finite automata: strengths, weaknesses and generalizations. In: Proceedings 39th Annual Symposium on Foundations of Computer Science (Cat. No. 98CB36280). pp. 332–341. IEEE (1998)
3. Anticoli, L., Piazza, C., Taglialegne, L., Zuliani, P.: Towards quantum programs verification: from quipper circuits to qpmc. In: International Conference on Reversible Computation. pp. 213–219. Springer (2016)
4. Arrighi, P.: An overview of quantum cellular automata. Natural Computing **18**(4), 885–899 (2019)
5. Bell, P.C., Hirvensalo, M.: Acceptance ambiguity for quantum automata. In: 44th International Symposium on Mathematical Foundations of Computer Science (MFCS 2019). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2019)
6. Belovs, A., Rosmanis, A., Smotrovs, J.: Multi-letter reversible and quantum finite automata. In: International Conference on Developments in Language Theory. pp. 60–71. Springer (2007)
7. Bergelson, V., Leibman, A.: Distribution of values of bounded generalized polynomials. Acta Mathematica **198**(2), 155–230 (2007)
8. Bertoni, A., Carpentieri, M.: Analogies and differences between quantum and stochastic automata. Theoretical Computer Science **262**(1-2), 69–81 (2001)
9. Bertoni, A., Mereghetti, C., Palano, B.: Quantum computing: 1-way quantum automata. In: International conference on developments in language theory. pp. 1–20. Springer (2003)
10. Bhatia, A.S., Kumar, A.: Quantum finite automata: survey, status and research directions. arXiv preprint arXiv:1901.07992 (2019)
11. Bhatia, A.S., Kumar, A.: On relation between linear temporal logic and quantum finite automata. Journal of Logic, Language and Information **29**(2), 109–120 (2020)
12. Bianchi, M.P., Mereghetti, C., Palano, B.: Quantum finite automata: Advances on bertoni's ideas. Theoretical Computer Science **664**, 39–53 (2017)
13. Birkan, U., Salehi, Ö., Olejar, V., Nurlu, C., Yakaryılmaz, A.: Implementing quantum finite automata algorithms on noisy devices. In: International Conference on Computational Science. pp. 3–16. Springer (2021)
14. Brodsky, A., Pippenger, N.: Characterizations of 1-way quantum finite automata. SIAM Journal on Computing **31**(5), 1456–1478 (2002)
15. Clarke, E.M., Henzinger, T.A., Veith, H., Bloem, R., et al.: Handbook of model checking, vol. 10. Springer (2018)
16. Dirac, P.A.M.: Lectures on quantum field theory. American Journal of Physics **37**, 233–233 (1969)
17. Feng, Y., Yu, N., Ying, M.: Model checking quantum markov chains. Journal of Computer and System Sciences **79**(7), 1181–1198 (2013)
18. Gainutdinova, A., Yakaryılmaz, A.: Unary probabilistic and quantum automata on promise problems. Quantum Information Processing **17**(2), 1–17 (2018)
19. Gay, S.J., Nagarajan, R., Papanikolaou, N.: Qmc: A model checker for quantum systems. In: International Conference on Computer Aided Verification. pp. 543–547. Springer (2008)
20. Hermanns, H., Herzog, U., Katoen, J.P.: Process algebra for performance evaluation. Theoretical computer science **274**(1-2), 43–87 (2002)

21. Katoen, J.P.: The probabilistic model checking landscape. In: Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science. pp. 31–45 (2016)
22. Khadiev, K., Khadieva, A., Ziatdinov, M., Mannapov, I., Kravchenko, D., Rivosh, A., Yamilov, R.: Two-way and one-way quantum and classical automata with advice for online minimization problems. Theoretical Computer Science (2022)
23. Kondacs, A., Watrous, J.: On the power of quantum finite state automata. Proceedings 38th Annual Symposium on Foundations of Computer Science pp. 66–75 (1997)
24. Kwiatkowska, M., Norman, G., Parker, D.: Probabilistic model checking: Advances and applications. In: Formal System Verification, pp. 73–121. Springer (2018)
25. Mereghetti, C., Palano, B.: Guest column: Quantum finite automata: From theory to practice. ACM SIGACT News **52**(3), 38–59 (2021)
26. Mereghetti, C., Palano, B., Cialdi, S., Vento, V., Paris, M.G., Olivares, S.: Photonic realization of a quantum finite automaton. Physical Review Research **2**(1), 013089 (2020)
27. Moore, C., Crutchfield, J.P.: Quantum automata and quantum grammars. Theoretical Computer Science **237**(1-2), 275–306 (2000)
28. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information (2002)
29. Paschen, K.: Quantum finite automata using ancilla qubits. Tech. rep., Universität Karlsruhe (TH) (2000). https://doi.org/10.5445/IR/1452000
30. Qiu, D., Yu, S.: Hierarchy and equivalence of multi-letter quantum finite automata. Theoretical Computer Science **410**(30-32), 3006–3017 (2009)
31. Qiu, D., Mateus, P., Sernadas, A.: One-way quantum finite automata together with classical states. arXiv preprint arXiv:0909.1428 pp. 3006–3017 (2009)
32. Rabin, M.O.: Probabilistic automata. Information and control **6**(3), 230–245 (1963)
33. Von Neumann, J.: Mathematical foundations of quantum mechanics. In: Mathematical Foundations of Quantum Mechanics. Princeton university press (2018)