

Quantum Markov chains: description of hybrid systems, decidability of equivalence, and model checking linear-time properties

Lvzhou Li^{a,b,d,*}, Yuan Feng^{b,c}

^a*Department of Computer Science, Sun Yat-sen University, Guangzhou 510006, China*

^b*Center for Quantum Computation and Intelligent Systems, FEIT, University of Technology Sydney, Sydney 2007, Australia*

^c*AMSS-UTS Joint Research Laboratory for Quantum Computation, Chinese Academy of Sciences, Beijing, China*

^d*The Guangdong Key Laboratory of Information Security Technology, Sun Yat-sen University, 510006, China*

Abstract

In this paper, we study a model of quantum Markov chains that is a quantum analogue of Markov chains and is obtained by replacing probabilities in transition matrices with quantum operations. We show that this model is very suited to describe hybrid systems that consist of a quantum component and a classical one, although it has the same expressive power as another quantum Markov model proposed in the literature. Indeed, hybrid systems are often encountered in quantum information processing; for example, both quantum programs and quantum protocols can be regarded as hybrid systems. Thus, we further propose a model called hybrid quantum automata (HQA) that can be used to describe these hybrid systems that receive inputs (actions) from the outer world. We show the language equivalence problem of HQA is decidable in polynomial time. Furthermore, we apply this result to the trace equivalence problem of quantum Markov chains, and thus it is also decidable in polynomial time. Finally, we discuss model checking linear-time properties of quantum Markov chains, and show the quantitative analysis of regular safety properties can be addressed successfully.

Keywords: Quantum Markov chains, hybrid systems, quantum automata, equivalence, model checking, linear-time property

*Corresponding author. Mobile: +8613802437672; Corresponding address: Department of Computer Science, Sun Yat-sen University, Guangzhou 510006, China

E-mail address: lilvzh@mail.sysu.edu.cn (L. Li); Yuan.Feng@uts.edu.au (Y. Feng)

1. Introduction

As we know, Markov chains as a mathematical model for stochastic systems play a fundamental role in computer science and even in the whole field of information science. A Markov chain is usually represented by a pair (P, π_0) where π_0 is a vector standing for the initial state of a stochastic system, and P is a stochastic matrix¹ characterizing the evolution of the system. Over the past two decades, quantum computing and quantum information have attracted considerable attention from the academic community. Then it is natural to study the quantum analogue of Markov chains. Actually, the terminology “quantum Markov chains” have appeared many times in the literature [1, 2, 9, 8, 17, 28], although it does not mean exactly the same thing in different references. A usual approach to defining quantum Markov chains is to view a quantum Markov chain as a pair (\mathcal{E}, ρ_0) where ρ_0 , a density operator, denotes an initial state of a quantum system, and \mathcal{E} is a trace-preserving quantum operation that characterizes the dynamics of the quantum system. This resembles very closely a classical Markov chain represented by a pair (P, π_0) . Indeed, in the textbook [19], when quantum operations were introduced, they were viewed as a quantum analogue of Markov processes. In [17, 28], a quantum Markov chain means the same thing as mentioned here, while it mainly means a quantum walk in [1].

In this paper, we focus on the quantum Markov model reported in [9, 8] which is greatly different from the one mentioned above but will be shown to be very suited to describe hybrid systems that consist of a quantum component and a classical one. Such a quantum Markov chain can be roughly represented by a pair (M, μ_0) where M is a transition matrix resembling P in a classical Markov chain but replacing each transition probability with a quantum operation and satisfying the condition that the sum of each column of M is a trace-preserving quantum operation. μ_0 , standing for the initial state of the model, is a vector with each entry being a density operator up to a factor. This model looks very strange at first glance, but it has the same expressive power as the conventional one given by (\mathcal{E}, ρ_0) . Specially, we will show that this model is very suited to describe hybrid systems that consists of a quantum component and a classical one. Indeed, hybrid systems are often encountered in quantum computing and quantum information, varying from quantum Turing machines [26] and quantum finite automata [14, 22] to quantum programs [24] and quantum protocols

¹In this paper, a matrix is said to be a stochastic matrix if each column of it is a probabilistic distribution.

such as BB84. Quantum engineering systems developed in the future will most probably have a classical human-interactive interface and a quantum processor, and thus they will be hybrid models. Therefore, it is worth developing a theory for describing and verifying hybrid systems.

In order to describe hybrid systems that receive inputs or actions from the outer world, we propose the notion of hybrid quantum automata (HQA) that generalize semi-quantum finite automata or other models studied by Ambainis and Watrous, and Qiu etc (see e.g. [3, 5, 21, 30, 31, 29]). In fact, these automata in the mentioned references as hybrid systems have been described in a uniform way by the authors [14]. When viewing HQA as language acceptors, we show their language equivalence problem is decidable in polynomial time by transforming this problem to the equivalence problem of probabilistic automata. Furthermore, we apply this result to the trace equivalence problem of quantum Markov chains, showing the trace equivalence problem is also decidable in polynomial time.

Finally, we consider model checking linear-time properties of hybrid systems that are modeled by quantum Markov chains. We show that the quantitative analysis of regular safety properties can be addressed as done for stochastic systems, by transforming it to the reachability problem that can be addressed by determining a least solution of a system of linear equations. For general ω -regular properties, the similar technical treatments used for stochastic systems no longer take effect for our purpose, and some new techniques need to be explored in the further study.

2. Preliminaries

A Hilbert space is usually denoted by the symbol \mathcal{H} . $\dim(\mathcal{H})$ stands for the dimension of \mathcal{H} . Let $\mathcal{L}(\mathcal{H})$ be the set of all linear operators from \mathcal{H} to itself. A^* , A^\dagger and A^\top denote respectively the conjugate, the conjugate-transpose, and the transpose of operator A . The trace of A is denoted by $\text{Tr}(A)$. $A \in \mathcal{L}(\mathcal{H})$ is said to be positive, denoted by $A \geq 0$, if $\langle \psi | A | \psi \rangle \geq 0$ for any $|\psi\rangle \in \mathcal{H}$. $A \geq B$ if $A - B$ is positive. Let

$$\mathcal{P}(\mathcal{H}) = \{A \in \mathcal{L}(\mathcal{H}) : A \geq 0\}.$$

Given a nonempty and countable set S , let

$$\text{Dist}_{\mathcal{H}}(S) = \{\mu : S \rightarrow \mathcal{P}(\mathcal{H}) : \sum_{s \in S} \text{Tr}(\mu(s)) = 1\}.$$

Elements in $\text{Dist}_{\mathcal{H}}(S)$ are called *positive-operator valued distributions*.

The detailed background on quantum information can be referred to the textbook [19] and lecture notes [27]. Here we just introduce briefly some necessary notions. States of a quantum system are described by density operators that are positive operators having unit trace. Let

$$\mathcal{D}(\mathcal{H}) = \{A \in \mathcal{P}(\mathcal{H}) : \text{Tr}(A) = 1\},$$

which denotes the set of all density operators on Hilbert space \mathcal{H} . An element in $\mathcal{D}(\mathcal{H})$ is generally indicated by the symbol ρ . A positive operator with trace less than 1 is called a *partial quantum state*.

A mapping $\mathcal{E}: \mathcal{L}(\mathcal{H}) \rightarrow \mathcal{L}(\mathcal{H})$ is called a *super-operator* on \mathcal{H} . \mathcal{E} is said to be *trace-preserving* if $\text{Tr}(\mathcal{E}(A)) = \text{Tr}(A)$ for all $A \in \mathcal{L}(\mathcal{H})$. Let $\mathcal{I}_{\mathcal{H}}$ and $\mathbf{0}_{\mathcal{H}}$ denote the identity and zero super-operators, respectively, and if \mathcal{H} is clear from the context the subscript \mathcal{H} is omitted. For two super-operators \mathcal{E} and \mathcal{F} , their summation, subtraction and multiplication, denoted by $\mathcal{E} + \mathcal{F}$, $\mathcal{E} - \mathcal{F}$ and $\mathcal{E} \circ \mathcal{F}$, respectively, are defined by

$$\begin{aligned} (\mathcal{E} + \mathcal{F})(A) &= \mathcal{E}(A) + \mathcal{F}(A), \\ (\mathcal{E} - \mathcal{F})(A) &= \mathcal{E}(A) - \mathcal{F}(A), \\ \mathcal{E} \circ \mathcal{F}(A) &= \mathcal{E}(\mathcal{F}(A)) \end{aligned}$$

for all $A \in \mathcal{L}(\mathcal{H})$. We always omit the symbol \circ and write $\mathcal{E}\mathcal{F}$ simply for $\mathcal{E} \circ \mathcal{F}$. The relation \approx between super-operators on \mathcal{H} is defined by: $\mathcal{E} \approx \mathcal{F}$ if $\text{Tr}(\mathcal{E}(\rho)) = \text{Tr}(\mathcal{F}(\rho))$ for all $\rho \in \mathcal{D}(\mathcal{H})$.

The evolution of a quantum system is characterized by *completely positive super-operators (CPOs)*. Here we do not recall the original definition of “completely positive”, but give an equivalent characterization. A super-operator $\mathcal{E}: \mathcal{L}(\mathcal{H}) \rightarrow \mathcal{L}(\mathcal{H})$ is said to be *completely positive* if and only if it has an *operator-sum representation (also called Kraus representation)* as

$$\mathcal{E}(A) = \sum_k E_k A E_k^\dagger,$$

where the set $\{E_k \in \mathcal{L}(\mathcal{H})\}$ are called *operation elements* of \mathcal{E} . \mathcal{E} is trace-preserving if and only if its operator-sum representation satisfies the following *completeness condition*

$$\sum_k E_k^\dagger E_k = I. \tag{1}$$

If Eq. (1) is replaced by $\sum_k E_k^\dagger E_k \leq I$, then \mathcal{E} is said to be *trace-nonincreasing*. By $\mathcal{SI}(\mathcal{H})$ we mean the set of all trace-nonincreasing CPOs on \mathcal{H} . Here the

reason why we used the notation $\mathcal{SI}(\mathcal{H})$ is to keep up with the one in [8]. Elements in $\mathcal{SI}(\mathcal{H})$ are called *quantum operations* [19]. Trace-preserving CPOs are called trace-preserving quantum operations.

Here we recall the concept of *selective quantum operations* [26] that are actually trace-preserving quantum operations equipped with a physical interpretation. A selective quantum operation is a mapping that takes as input $\rho \in \mathcal{D}(\mathcal{H})$ and outputs a probability over pairs of the form (τ, ρ_τ) , where $\tau \in \Delta$ and $\rho_\tau \in \mathcal{D}(\mathcal{H})$. We refer to τ as the classical output of the operation; this may be the result of some measurement performed on ρ , but this is not the most general situation. A selective quantum operation \mathcal{E} is described by a set of operation elements

$$\{E_{\tau,k} : \tau \in \Delta, k \in K_\tau\}$$

which satisfy the completeness condition $\sum_{\tau \in \Delta} \sum_{k \in K_\tau} E_{\tau,k}^\dagger E_{\tau,k} = I$. For each $\tau \in \Delta$, we defined Φ_τ as follows:

$$\Phi_\tau(\rho) = \sum_{k \in K_\tau} E_{\tau,k} \rho E_{\tau,k}^\dagger.$$

Then \mathcal{E} is represented by $\mathcal{E} = \{\Phi_\tau : \tau \in \Delta\}$. For $\tau \in \Delta$, let $p_\tau = \text{Tr}(\Phi_\tau(\rho))$ and $\rho_\tau = \Phi_\tau(\rho)/p_\tau$ (in case $p_\tau = 0$, ρ_τ is undefined). Now, on input ρ , the output of \mathcal{E} is defined to be (τ, ρ_τ) with probability p_τ .

In the following, we recall a useful linear mapping *vec* from [27] which maps a matrix $A \in \mathbb{C}^{n \times n}$ to an n^2 -dimensional column vector, defined as follows:

$$\text{vec}(A)((i-1)n+j) = A(i,j).$$

In other words, $\text{vec}(A)$ is the vector obtained by taking the rows of A , transposing them to form column vectors, and stacking those column vectors on top of one another to form a single vector. For example, we have

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \text{ and } \text{vec}(A) = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}.$$

If we let $|i\rangle$ be an n -dimensional column vector with the i th entry being 1 and else 0's, then $\{|i\rangle\langle j| : i, j = 1, \dots, n\}$ form a basis of $\mathbb{C}^{n \times n}$. Therefore, the mapping *vec* can also be defined as $\text{vec}(|i\rangle\langle j|) = |i\rangle|j\rangle$.

Let A, B, C be $n \times n$ matrices. Then we have

$$\text{vec}(AXB) = (A \otimes B^\top) \text{vec}(X), \quad (2)$$

$$\text{Tr}(AB) = \text{vec}(A^\top)^\top \text{vec}(B). \quad (3)$$

Throughout this paper, we use $|S|$ to denote the cardinality of the set S .

3. Quantum Markov chains and hybrid systems

As mentioned in Introduction, it is natural to propose the quantum analogue of Markov chains, while Markov chains have been shown to play a fundamental role in information science and quantum information processing has attracted more and more attention from the academic community. In the literature, there are several notions of quantum Markov chains defined from different perspectives, but in this paper we focus mainly on the one reported in [9, 8]. In the following, we show that the quantum Markov model given in [9, 8] is very suited to describe hybrid systems, although it has the same expressive power as the conventional one given in [17, 28]. Before that, we recall some necessary definitions below.

One natural viewpoint is to regard a quantum Markov chain as a pair of a trace-preserving quantum operation and a density operator. Here we keep up with the notations used in [28].

Definition 1. *A quantum Markov chain (qMC) is a triple $\mathcal{M} = (\mathcal{H}, \mathcal{E}, \rho_0)$ where:*

- \mathcal{H} is a Hilbert space;
- \mathcal{E} is a trace-preserving quantum operation over \mathcal{H} ;
- ρ_0 is the initial density operator over \mathcal{H} .

\mathcal{M} is said to be finite if $\dim(\mathcal{H})$ is finite.

The physical interpretation of qMC \mathcal{M} is: a quantum system with Hilbert space \mathcal{H} starts in the initial state ρ_0 , and at each step the state evolves according to \mathcal{E} . Usually, we do not give explicitly the underlying Hilbert space \mathcal{H} , and thus a qMC is simply denoted by the pair (\mathcal{E}, ρ_0) . The state at the n th step is denoted by ρ_n . Then we have

$$\rho_n = \mathcal{E}^n(\rho_0) \tag{4}$$

where \mathcal{E}^n is inductively defined by: (i) $\mathcal{E}^0 = \mathcal{I}$ and (ii) $\mathcal{E}^n = \mathcal{E}\mathcal{E}^{n-1}$ for $n = 1, 2, \dots$.

Another quantum analogue of Markov chains was reported in [9, 8], and we call it *hybrid quantum Markov chain (hqMC)*. The reason why we called them “hybrid” will get clear soon. The definition is as follows.

Definition 2. An hqMC is represented by a tuple $\mathcal{M} = (\mathcal{H}, S, M, \mu_0)$ where:

- \mathcal{H} is a Hilbert space;
- S is a nonempty and countable set of states;
- $M : S \times S \rightarrow \mathcal{SI}(\mathcal{H})$ such that $\sum_{t \in S} M(t, s)$ is a trace-preserving quantum operation for each $s \in S$.
- $\mu_0 \in \text{Dist}_{\mathcal{H}}(S)$ denotes the initial distribution.

\mathcal{M} is said to be finite, if S and $\dim(\mathcal{H})$ are finite.

Visually, hqMC \mathcal{M} can be represented by a transition graph (a digraph) where states from S act as vertexes and there is an edge from s to t with label $M(t, s)$ if and only if $M(t, s) \neq \mathbf{0}$. For example, Fig. 1 represents an hqMC whose state set is $S = \{s_0, s_1, s_2\}$ and whose transition function M is given by $M(s_1, s_0) = \mathcal{E}_{01}, M(s_2, s_0) = \mathcal{E}_{02}, M(s_2, s_2) = \mathcal{E}_{22}, M(s_1, s_2) = \mathcal{E}_{21}, M(s_1, s_1) = \mathcal{I}$ where all given quantum operations are nonzero.

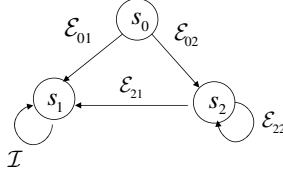


Figure 1: Transition graph of an hqMC.

In the sequel, we identify the transition function M with a $|S| \times |S|$ matrix of which each entry is a quantum operation and the sum of each column is a trace-preserving quantum operation. $M(t, s)$ denotes the entry in the t th row and the s th column. Similarly, μ_0 is viewed as a $|S|$ -dimensional column vector with each entry being a positive operator and their sum being a density operator. $\mu_0(s)$ denotes the s th entry. A rough interpretation of hqMC \mathcal{M} is: the system first starts in μ_0 and then at each step evolves according to M . At the n th step, the state is denoted by

$$\mu_n = M^n \mu_0$$

where M^n is inductively defined by: i) M^0 is a diagonal matrix with the diagonal entries being \mathcal{I} and other entries being $\mathbf{0}$, and ii) $M^n = M M^{n-1}$. Note that the multiplication of an entry in M and an entry in μ_n is in the sense of performing a

quantum operation on a partial quantum state. For example, for $\mu_1 = M\mu_0$ we have $\mu_1(s) = \sum_{t \in S} M(s, t)(\mu_0(t))$ for $s \in S$ where quantum operation $M(s, t)$ is performed on the state $\mu_0(t)$.

In the following, we show that the model of hqMC $\mathcal{M} = (\mathcal{H}, S, M, \mu_0)$ is very suited to describe hybrid systems that are dynamic systems consisting of two interactive components: a quantum one and a classical one, although this might not be clearly noticed when the model was proposed at the beginning. As shown in Fig. 2, there is a hybrid system consisting of a quantum component whose state space is \mathcal{H} and a classical component whose state set is S . The behavior of this system is exactly described by \mathcal{M} . More specifically, at each step the hybrid system evolves as follows.

- (i) Firstly, depending on the current classical state s , quantum state ρ evolves according to M_s . M_s denotes the selective quantum operation described by the s th column of M , that is, $M_s = \{M(t, s) : t \in S\}$. Thus, on input ρ , the output is $(t, M(t, s)(\rho)/p_t)$ with probability $p_t = \text{Tr}(M(t, s)(\rho))$.
- (ii) Secondly, the classical state s evolves into state t , where t is the output of the above quantum evolution. Equivalently, a transition function $\mathcal{F}_t : S \rightarrow S$ changes each state to t . If S is finite, then the classical component can be viewed as a DFA whose state set and input alphabet are both S and whose transition function maps the current state to the state indicated by the current input symbol.

The initial state of the hybrid system is $(s, \mu_0(s)/p_s)$ with probability $p_s = \text{Tr}(\mu_0(s))$ where $s \in S$ denotes the classical state.

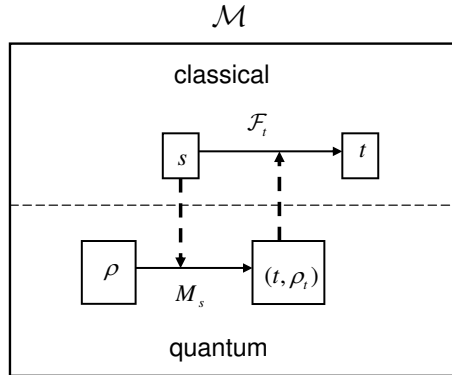


Figure 2: An hqMC describing the behavior of a hybrid system.

As shown above, the hqMC model is suited to describe hybrid systems. Indeed, hybrid systems are often encountered in quantum information processing. For example, quantum programs can be regarded as hybrid systems, since as stated by Selinger [24], quantum programs can be described by “quantum data with classical control flows”. Quantum data are represented by states of the quantum component and classical control flows are state evolutions of the classical component. Also note that the quantum Turing machines defined in [26] and quantum finite automata studied in [14] are all hybrid systems. In addition, as shown in [8], quantum cryptographic protocols such as BB84 protocol can be described by hqMC. We think that hybrid systems will be encountered more often as the study of quantum information goes ahead. In fact, since what we can observe are classical, the quantum engineering systems developed in the future will most probably have a classical human-interactive interface and a quantum processor, and thus they will be hybrid systems.

One should distinguish “hybrid systems” in this paper from those in [10, 23]. Hybrid systems in [10, 23] are digital real-time systems embedded in analog environments. Those systems combine discrete and continuous dynamics. There have been a long list of publications devoted to the verification of these hybrid systems. Hybrid systems in our paper are such systems that combine classical discrete dynamics and quantum discrete dynamics (it is also possible to consider quantum continuous dynamics), and in the sequel, when mentioning “hybrid systems” we always adopt this meaning. As mentioned above, hybrid systems often present in quantum information processing. Therefore, it could be meaningful and interesting to develop a theory of describing and verifying these systems; Feng et al.’s work [8] can be seen as a first step toward this direction.

In the following, we clarify the relationship between the two models of quantum Markov chains presented in this section. First, a qMC is obviously a special hqMC, since when there is only one classical state in an hqMC, it reduces to a qMC. On the other hand, we will show that each hqMC can also be simulated by a qMC. This is formally expressed in the following theorem.

Theorem 1. *Given an hqMC $\mathcal{M} = (\mathcal{H}, S, M, \mu_0)$, there exists a qMC $\mathcal{M}' = (\mathcal{H}', \mathcal{E}, \rho_0)$ such that the states ρ_n and μ_n of \mathcal{M}' and \mathcal{M} at the n th step, respectively, satisfy:*

$$\rho_n = \sum_{s \in S} |s\rangle\langle s| \otimes \mu_n(s)$$

for $n = 0, 1, \dots$.

Proof. The qMC $\mathcal{M}' = (\mathcal{H}', \mathcal{E}, \rho_0)$ is constructed as follows:

- $\mathcal{H}' = \mathcal{H}_S \otimes \mathcal{H}$ where $\mathcal{H}_S = \text{span}\{|s\rangle : s \in S\}$;
- $\rho_0 = \sum_{s \in S} |s\rangle\langle s| \otimes \mu_0(s)$;
- \mathcal{E} is a trace-preserving quantum operation on \mathcal{H}' which is constructed to simulate the interactive actions between the quantum component and the classical one in Fig. 2.

More specifically, \mathcal{E} is described by the set of operation elements

$$\{|t\rangle\langle s| \otimes M_{ts}^k : s, t \in S, k \in K_{ts}\} \quad (5)$$

where for each pair $t, s \in S$, $\{M_{ts}^k : k \in K_{ts}\}$ are operation elements of the quantum operation $M(t, s)$.

Then \mathcal{E} is trace-preserving since we have

$$\begin{aligned} & \sum_{s, t \in S} \sum_{k \in K_{ts}} (|t\rangle\langle s| \otimes M_{ts}^k)^\dagger (|t\rangle\langle s| \otimes M_{ts}^k) \\ &= \sum_{s, t \in S} \left(|s\rangle\langle s| \otimes \sum_{k \in K_{ts}} (M_{ts}^k)^\dagger M_{ts}^k \right) \\ &= \sum_{s \in S} \left(|s\rangle\langle s| \otimes \sum_{t \in S} \sum_{k \in K_{ts}} (M_{ts}^k)^\dagger M_{ts}^k \right) \\ &\stackrel{(a)}{=} \sum_{s \in S} |s\rangle\langle s| \otimes I_{\mathcal{H}} = I_{\mathcal{H}_S} \otimes I_{\mathcal{H}} \end{aligned}$$

where $I_{\mathcal{H}_S}$ and $I_{\mathcal{H}}$ denote identity operators on \mathcal{H}_S and \mathcal{H} , respectively. In the above, equation (a) holds because $\{M_{ts}^k : k \in K_{ts}, t \in S\}$ are operation elements of the selective quantum operation M_s .

Furthermore, for $\rho \otimes \varrho \in \mathcal{L}(\mathcal{H}_S \otimes \mathcal{H})$, by a direct calculation we have

$$\mathcal{E}(\rho \otimes \varrho) = \sum_{s, t \in S} \langle s|\rho|s\rangle |t\rangle\langle t| \otimes M(t, s)(\varrho) \quad (6)$$

$$= \sum_{s, t \in S} \langle s|\rho|s\rangle \mathcal{F}_t(|s\rangle\langle s|) \otimes M(t, s)(\varrho). \quad (7)$$

From the above, it can be seen that the intuitive idea of \mathcal{E} is as follows: i) first perform the measurement $\{E_s = |s\rangle\langle s| : s \in S\}$ on the classical system to observe its state; ii) if s is the result, then perform the selective quantum operation $M_s = \{M(t, s) : t \in S\}$ on the quantum system; iii) if the classical output of M_s is t , then perform \mathcal{F}_t on the classical system changing its state to

t . Note that \mathcal{F}_t is a trace-preserving quantum operation for each $t \in S$, since it has operation elements $\{F_t^s = |t\rangle\langle s|, s \in S\}$ that satisfy the completeness condition.

Now by induction on n we prove $\rho_n = \sum_{s \in S} |s\rangle\langle s| \otimes \mu_n(s)$ for $n = 0, 1, \dots$. First when $n = 0$, we have $\rho_0 = \sum_{s \in S} |s\rangle\langle s| \otimes \mu_0(s)$. Suppose it holds for n . Then from Eq. (6) we have

$$\begin{aligned} \rho_{n+1} &= \mathcal{E}(\rho_n) = \mathcal{E}\left(\sum_{t \in S} |t\rangle\langle t| \otimes \mu_n(t)\right) \\ &= \sum_{s, s' \in S} \sum_{t \in S} \langle s|t\rangle\langle t|s\rangle |s'\rangle\langle s'| \otimes M(s', s)(\mu_n(t)) \\ &= \sum_{s' \in S} |s'\rangle\langle s'| \otimes \sum_{t \in S} M(s', t)(\mu_n(t)) \\ &= \sum_{s' \in S} |s'\rangle\langle s'| \otimes \mu_{n+1}(s'). \end{aligned}$$

Thus, we have completed the proof. \square

From the above discussion, we know that qMC and hqMC have the same expressive power. However, they are suitable for describing different systems. While it is natural to describe a purely quantum system using the qMC model, it is convenient to describe a hybrid system using the hqMC model.

4. Hybrid quantum automata

Based on the hqMC model, we propose an automaton model—hybrid quantum automata, which generalizes the models in [14] and can be used to describe hybrid systems that receive inputs (or actions) from the outer world.

First, some notations are explained below. As usual, for nonempty set Σ , by Σ^* we mean the set of all finite-length strings over Σ . Let $\Sigma^+ = \Sigma^* \setminus \{\epsilon\}$ where ϵ denotes the empty string. For $u \in \Sigma^*$, $|u|$ denotes the length of u . Let $\Sigma^n = \{u \in \Sigma^* : |u| = n\}$ and $\Sigma^{\leq n} = \{u \in \Sigma^* : |u| \leq n\}$.

Definition 3. A hybrid quantum automaton (HQA) is a tuple

$$\mathcal{A} = (\mathcal{H}, S, \Sigma, \mu_0, \{M_\sigma\}_{\sigma \in \Sigma})$$

where

- \mathcal{H} is a Hilbert space;
- S is a countable nonempty set of states;

- Σ is an alphabet of symbols;
- $\mu_0 \in \text{Dist}_{\mathcal{H}}(S)$ is the initial distribution;
- For each $\sigma \in \Sigma$, $M_\sigma : S \times S \rightarrow \mathcal{SI}(\mathcal{H})$ such that $\sum_{t \in S} M_\sigma(t, s)$ is a trace-preserving quantum operation for each $s \in S$;

\mathcal{A} is said to be finite, if S , Σ and $\dim(\mathcal{H})$ are finite.

The behavior of \mathcal{A} is roughly as: \mathcal{A} starts in μ_0 , and at each step, it scans the current input symbol $\sigma \in \Sigma$, and then updates its state according to M_σ . In this paper, we regard HQA \mathcal{A} as a language acceptor, that is, for each input $w \in \Sigma^*$, \mathcal{A} observes its final state after scanning all input symbols and accepts if the final state satisfies some given property. Generally, the accepting behavior is probabilistic, because of the inherent probabilism of quantum mechanics.

Here, we have two basic approaches to defining the automaton's accepting fashions. One is based on classical states: \mathcal{A} accepts its input $w \in \Sigma^*$, if its classical state after scanning the whole input belongs to a subset $F \subseteq S$. In this case, the model is represented by a tuple $\mathcal{A} = (\mathcal{H}, S, \Sigma, \mu_0, \{M_\sigma\}_{\sigma \in \Sigma}, F)$, \mathcal{A} is said to *accept with classical fashion*, and we call it a C-HQA for short. Then C-HQA \mathcal{A} defines a function $P_{\mathcal{A}} : \Sigma^* \rightarrow [0, 1]$ as

$$P_{\mathcal{A}}(w) = \sum_{s \in F} \text{Tr}((M_w \mu_0)(s))$$

where $M_{\sigma_1 \sigma_1 \dots \sigma_n} = M_{\sigma_n} \dots M_{\sigma_2} M_{\sigma_1}$ and M_ϵ is a diagonal matrix with the diagonal entries being \mathcal{I} and others being $\mathbf{0}$. $P_{\mathcal{A}}(w)$ denotes the probability that \mathcal{A} accepts w .

Also, we can define that HQA \mathcal{A} accepts its input if its final quantum state belongs to a subspace of \mathcal{H} , say \mathcal{H}_{acc} . Let P_{acc} be the projector onto \mathcal{H}_{acc} . In this case, the model is given by $\mathcal{A} = (\mathcal{H}, S, \Sigma, \mu_0, \{M_\sigma\}_{\sigma \in \Sigma}, P_{acc})$, \mathcal{A} is said to *accept with quantum fashion*, and we call it a Q-HQA for short. The probability that \mathcal{A} accepts its input $w \in \Sigma^*$ is given by

$$P_{\mathcal{A}}(w) = \sum_{s \in S} \text{Tr}(P_{acc}(M_w \mu_0)(s)).$$

Based on the above two basic accepting fashions, HQA can generally have a mixed accepting fashion. In this case, the model is called M-HQA for short and is represented by $\mathcal{A} = (\mathcal{H}, S, \Sigma, \mu_0, \{M_\sigma\}_{\sigma \in \Sigma}, F, P_{acc})$. The accepting probability on input $w \in \Sigma^*$ is give by

$$P_{\mathcal{A}}(w) = \sum_{s \in F} \text{Tr}(P_{acc}(M_w \mu_0)(s)).$$

Remark 1. (i) It is obvious that a probabilistic automaton is a degenerate C-HQA in which $M(s, t) = p_{s,t}\mathcal{I}$ for all $s, t \in S$ with $(p_{s,t})_{s,t \in S}$ being a stochastic matrix, and $\mu_0(s) = p_s \rho$ for some density operator ρ and $s \in S$ with $(p_s)_{s \in S}$ being a probabilistic distribution. (ii) Note that we characterized three models of quantum finite automata in the framework of hybrid systems in [14]; all of them can be regarded as a concrete implementation of the HQA model defined in this paper. For example, CL-1QFA [5] and 1QCFA [30] are instances of C-HQA, and 1QFAC [21] are instances of Q-HQA. Thus, the HQA given by us is a generalized model.

Associated with the qMC model, there is another quantum automaton model that was studied in [15, 11].

Definition 4. A quantum automaton (QA) is a tuple $\mathcal{A} = (\mathcal{H}, \Sigma, \rho_0, \{\mathcal{E}_\sigma\}_{\sigma \in \Sigma}, P_{acc})$ where \mathcal{H} is a Hilbert space, Σ is an alphabet, $\rho_0 \in \mathcal{D}(\mathcal{H})$ is the initial state, \mathcal{E}_σ is a trace-preserving quantum operation for each $\sigma \in \Sigma$, P_{acc} denotes a projector onto a subspace of \mathcal{H} (called an accepting subspace). \mathcal{A} is said to be finite, if Σ and $\dim(\mathcal{H})$ are finite. For each input $w = \sigma_1 \cdots \sigma_k \in \Sigma^*$, the accepting probability is given by $P_{\mathcal{A}}(w) = \text{Tr}(P_{acc} \mathcal{E}_w(\rho_0))$ where $\mathcal{E}_w = \mathcal{E}_{\sigma_k} \cdots \mathcal{E}_{\sigma_1}$.

Implied by Theorem 1, we have the following result.

Lemma 1. For each HQA \mathcal{A} over alphabet Σ , there is a QA \mathcal{A}' such that $P_{\mathcal{A}}(w) = P_{\mathcal{A}'}(w)$ for all $w \in \Sigma^*$.

Proof. The idea is similar to the procedure of simulating hqMC by qMC, and we sketch it as follows. Given an HQA $\mathcal{A} = (\mathcal{H}, S, \Sigma, \mu_0, \{M_\sigma\}_{\sigma \in \Sigma})$, we construct a QA $\mathcal{A}' = (\mathcal{H}', \Sigma, \rho_0, \{\mathcal{E}_\sigma\}_{\sigma \in \Sigma}, P'_{acc})$ where $\mathcal{H}' = \mathcal{H}_S \otimes \mathcal{H}$, $\rho_0 = \sum_{s \in S} |s\rangle\langle s| \otimes \mu_0(s)$, and for each $\sigma \in \Sigma$, \mathcal{E}_σ is constructed from M_σ as done in Theorem 1. Let $\rho_w = \mathcal{E}_w(\rho_0)$ and $\mu_w = M_w \mu_0$ with $w \in \Sigma^*$. Then as shown in Theorem 1 we have $\rho_w = \sum_{s \in S} |s\rangle\langle s| \otimes \mu_w(s)$. The last step is to construct P'_{acc} , which is dependent on the accepting fashion of \mathcal{A} :

- (a) If \mathcal{A} is a C-HQA and assume that its classical accepting set is $F \subseteq S$, then we let $P'_{acc} = \sum_{s \in F} |s\rangle\langle s| \otimes I_{\mathcal{H}}$.
- (b) If \mathcal{A} is a Q-HQA with projector is P_{acc} , then we let $P'_{acc} = I_{\mathcal{H}_S} \otimes P_{acc}$.
- (c) If \mathcal{A} is an M-HQA, then let $P'_{acc} = \sum_{s \in F} |s\rangle\langle s| \otimes P_{acc}$.

In any case, it is easy to verify that $P_{\mathcal{A}}(w) = P_{\mathcal{A}'}(w)$ for all $w \in \Sigma^*$. \square

Remark 2. From the above result it follows that HQA do not surpass QA in the sense of language recognition power. Note that it has been shown that finite QA recognize with bounded error exactly the family of regular language[15].

In the following we introduce another model that was called bilinear machine in [16].

Definition 5. A bilinear machine (BLM) is a tuple $\mathcal{A} = (n, \Sigma, \{M_\sigma\}_{\sigma \in \Sigma}, \pi, \eta)$ where $n \in \mathbb{N}$ is called the state number of \mathcal{A} , Σ is a finite alphabet, $M_\sigma \in \mathbb{C}^{n \times n}$ is a transition matrix for each $\sigma \in \Sigma$, $\pi \in \mathbb{C}^n$ is a column vector, and $\eta \in \mathbb{C}^n$ is a row vector. Automaton \mathcal{A} assigns each $w = \sigma_1 \cdots \sigma_k \in \Sigma^*$ a weight $P_{\mathcal{A}}(w)$ as $P_{\mathcal{A}}(w) = \eta M_{\sigma_k} \cdots M_{\sigma_1} \pi$. \mathcal{A} is a probabilistic automaton if it is further required that each M_σ is a stochastic matrix, π is a probabilistic distribution, and η has entries being 1 or 0.

Every finite QA can be simulated by a BLM, which is stated formally as follows.

Lemma 2. For each finite QA \mathcal{A} over alphabet Σ , there is a BLM \mathcal{A}' such that $P_{\mathcal{A}'}(w) = P_{\mathcal{A}}(w)$ for all $w \in \Sigma^*$.

Proof. Let finite QA $\mathcal{A} = (\mathcal{H}, \Sigma, \rho_0, \{\mathcal{E}_\sigma\}_{\sigma \in \Sigma}, P_{acc})$. For each $\sigma \in \Sigma$, suppose that $\mathcal{E}_\sigma(\rho) = \sum_k E_k^\sigma \rho E_k^{\sigma \dagger}$, and denote

$$A_\sigma = \sum_k E_k^\sigma \otimes E_k^{\sigma*}.$$

Then by Eq. (2), we have

$$\begin{aligned} \text{vec}(\mathcal{E}_{\sigma_1}(\rho)) &= A_{\sigma_1} \text{vec}(\rho), \\ \text{vec}(\mathcal{E}_{\sigma_2} \mathcal{E}_{\sigma_1}(\rho)) &= A_{\sigma_2} A_{\sigma_1} \text{vec}(\rho). \end{aligned}$$

As a result, the probability of \mathcal{A} accepting $w = \sigma_1 \cdots \sigma_k \in \Sigma^*$ can be rewritten in the following:

$$\begin{aligned} P_{\mathcal{A}}(w) &= \text{Tr}(P_{acc} \mathcal{E}_{\sigma_k} \cdots \mathcal{E}_{\sigma_1}(\rho_0)) \\ &= \text{vec}(P_{acc})^\top \text{vec}(\mathcal{E}_{\sigma_k} \cdots \mathcal{E}_{\sigma_1}(\rho_0)) \\ &= \text{vec}(P_{acc})^\top A_{\sigma_k} \cdots A_{\sigma_2} A_{\sigma_1} \text{vec}(\rho_0) \end{aligned}$$

where the second equality follows from Eq. (3). Therefore, we construct BLM $\mathcal{A}' = (n, \Sigma, \{M_\sigma\}_{\sigma \in \Sigma}, \alpha, \eta)$ with $n = \dim(\mathcal{H})^2$, $M_\sigma = A_\sigma$ for $\sigma \in \Sigma$, $\pi = \text{vec}(\rho_0)$, and $\eta = \text{vec}(P_{acc})^\top$. \square

In classical automata theory, it is a fundamental problem to decide whether two probabilistic automata have the same accepting probability for each input (that is known as the equivalence problem) [20, 25, 13]. This problem has some nontrivial applications; for example, [6] applied it to verification of equivalence between processes, and [18, 12] applied it to verification of equivalence between probabilistic programs. Taking account into that the model of HQA is suited to describe hybrid systems (including quantum programs), it is meaningful to consider the equivalence problem for HQA. Formally, the equivalence problem is as follows.

Definition 6. *Two HQA (QA, BLM) \mathcal{A}_1 and \mathcal{A}_2 over the same alphabet Σ are k -equivalent, if $P_{\mathcal{A}_1}(w) = P_{\mathcal{A}_2}(w)$ for all $w \in \Sigma^{\leq k}$. Furthermore, they are said to be equivalent if $P_{\mathcal{A}_1}(w) = P_{\mathcal{A}_2}(w)$ holds for all $w \in \Sigma^*$.*

The history for the equivalence problem of probabilistic automata is as follows. Paz [20] proved that two probabilistic automata are equivalent if and only if they are $(n_1 + n_2 - 1)$ -equivalent, where n_1 and n_2 are state numbers of the two automata. Afterwards, this result was improved by Tzeng [25] who proposed a polynomial-time algorithm determining whether two given probabilistic automata are equivalent or not, and the time complexity is $O(|\Sigma|(n_1 + n_2)^4)$. Recently, an improved complexity $O(|\Sigma|(n_1 + n_2)^3)$ was reported in [13]. As mentioned in [16], all these results are based on some ordinary knowledge about matrices and linear spaces rather than on any essential property of probabilistic automata; as a result, they also hold for BLM. We summarize these results as follows.

Lemma 3. *Two BLM \mathcal{A}_1 and \mathcal{A}_2 over Σ are equivalent if and only if they are $(n_1 + n_2 - 1)$ -equivalent, and there exists a $O(|\Sigma|(n_1 + n_2)^3)$ time algorithm deciding whether they are equivalent or not, where n_1 and n_2 are state numbers of \mathcal{A}_1 and \mathcal{A}_2 , respectively.*

For the sake of completeness, we present an algorithm for BLM's equivalence problem in Algorithm 1. For algorithmic purposes we assume that all inputs consist of complex numbers whose real and imaginary parts are rational numbers and that each arithmetic operation on rational numbers can be done in constant time. Let \mathcal{Q} be the set of vectors that have been added into *queue*, and let

$$\mathcal{S} = \text{span} \left\{ \begin{pmatrix} M_w^1 \pi_1 \\ M_w^2 \pi_2 \end{pmatrix} : w \in \Sigma^* \right\}$$

where $M_w^1 = M_{w_k}^1 \cdots M_{w_2}^1 M_{w_1}^1$ for $w = w_k \cdots w_2 w_1$ and it is similar for M_w^2 . Then the relationship among \mathcal{B} , \mathcal{Q} and \mathcal{S} is: \mathcal{Q} can be proved to be a basis for

\mathcal{S} , and \mathcal{B} is obtained from \mathcal{Q} by the Gram-Schmidt procedure and thus is an orthonormal basis for \mathcal{S} . Therefore, \mathcal{A}_1 and \mathcal{A}_2 are equivalent (i.e., $\eta_1 u_1 = \eta_2 u_2$ holds for all elements $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \in \mathcal{S}$) if and only if $\eta_1 u_1 = \eta_2 u_2$ for all elements $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \in \mathcal{Q}$. Let $n = n_1 + n_2$. Then $|\mathcal{Q}| = |\mathcal{B}|$ is at most n and the procedure $b := u - \sum_{b_i \in \mathcal{B}} (u^* b_i) b_i$ takes at most $O(n^2)$ time. The total time complexity is thus $O(|\Sigma|n^3)$.

Input: $\mathcal{A}_i = (n_i, \Sigma, \{M_\sigma^i\}_{\sigma \in \Sigma}, \pi_i, \eta_i)$ for $i = 1, 2$.

Output: \mathcal{A}_1 and \mathcal{A}_2 are equivalent or not.

$\mathcal{B} := \emptyset$; $queue := \emptyset$; $\pi := \begin{pmatrix} \pi_1 \\ \pi_2 \end{pmatrix}$;

If $\eta_1 \pi_1 \neq \eta_2 \pi_2$ **then**

 return “ \mathcal{A}_1 and \mathcal{A}_2 are not equivalent”;

If $\|\pi\| = 0$ **then**

 return “ \mathcal{A}_1 and \mathcal{A}_2 are equivalent”;

$\mathcal{B} := \{\frac{\pi}{\|\pi\|}\}$; add π to $queue$;

while $queue \neq \emptyset$ **do**

begin take $\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$ from $queue$;

 for all $\sigma \in \Sigma$ **do**

begin

$u_1 = M_\sigma^1 v_1$; $u_2 = M_\sigma^2 v_2$;

if $\eta_1 u_1 \neq \eta_2 u_2$ **then**

 return “ \mathcal{A}_1 and \mathcal{A}_2 are not equivalent”;

$u := \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$;

$b := u - \sum_{b_i \in \mathcal{B}} (u^* b_i) b_i$;

if $\|b\| \neq 0$ **then**

 add u to $queue$;

$\mathcal{B} := \mathcal{B} \cup \{\frac{b}{\|b\|}\}$;

end;

end;

 return “ \mathcal{A}_1 and \mathcal{A}_2 are equivalent”;

Algorithm 1: Determining whether two BLM are equivalent or not.

Remark 3. In Definition 6, if it is required that $P_{\mathcal{A}_1}(w) = P_{\mathcal{A}_2}(w)$ holds for

all $w \in \Sigma^+$ instead of for all $w \in \Sigma^*$, then the algorithm almost keeps the same and the complexity has no change. This case will be used in the next section for the trace equivalence problem of quantum Markov chains.

Now it follows from Lemmas 1, 2 and 3 that the equivalence problem of finite HQA is decidable in polynomial time.

Theorem 2. *Two finite HQA $\mathcal{A}_i = (\mathcal{H}^{(i)}, S^{(i)}, \mu_0^{(i)}, \Sigma, \{M_\sigma^{(i)}\}_{\sigma \in \Sigma})$ ($i = 1, 2$) are equivalent if and only if they are $((n_1 k_1)^2 + (n_2 k_2)^2 - 1)$ -equivalent where $n_i = \dim(\mathcal{H}^{(i)})$ and $k_i = |S^{(i)}|$. Furthermore, there exists a polynomial-time algorithm deciding whether they are equivalent or not.*

In the next section, we will show that the trace equivalence problem of quantum Markov chains can be transformed in linear time to the equivalence problem of QHA, and thus is also decidable in polynomial time.

5. Trace equivalence of quantum Markov chains

Feng et al [8] used the model of hqMC for model-checking quantum protocols where the purpose is to check whether the classical component of a hybrid system satisfies some given property. To that end, a labeling function was used to associate each classical state a set of atomic propositions that are satisfied at that state. In this paper, we called such an hqMC equipped with a state labeling function a *state-labeled hybrid quantum Markov chain* (SL-hqMC, for short). Also, we require that the hqMC is finite, although finiteness is not a necessary requirement for a general definition. The formal definition is as follows.

Definition 7. *A state-labeled hybrid quantum Markov chain (SL-hqMC) is a tuple*

$$\mathcal{M} = (\mathcal{H}, S, M, \mu_0, AP, L),$$

where

1. $(\mathcal{H}, S, M, \mu_0)$ is a finite hqMC;
2. AP is a finite set of atomic propositions;
3. $L : S \rightarrow 2^{AP}$ is a labeling function. L can be extended to finite sequence of states as $L(s_0 s_1 \dots s_n) = L(s_0) L(s_1) \dots L(s_n)$.

In the sequel, for the sake of simplicity we let $\Sigma = 2^{AP}$, and call it a labeling set. Then L assigns each state $s \in S$ a symbol $\sigma \in \Sigma$. For $\bar{s} = s_0 s_1 \dots s_k \in S^+$,

let

$$\rho_{\bar{s}} = \prod_{i=1}^k M(s_i, s_{i-1})(\mu_0(s_0)) \quad (8)$$

where $\prod_{i=1}^n A_i = A_n \cdots A_2 A_1$. Then $\text{Tr}(\rho_{\bar{s}})$ gives the probability of visiting the sequence of states $s_0 s_1 \cdots s_k$ when \mathcal{M} starts in the initial distribution μ_0 . Thus, SL-hqMC \mathcal{M} defines a function $P_{\mathcal{M}} : \Sigma^+ \rightarrow [0, 1]$ by

$$P_{\mathcal{M}}(w) = \sum_{\bar{s}: L(\bar{s})=w} \text{Tr}(\rho_{\bar{s}}).$$

This gives the probability of observing $w \in \Sigma^+$ when \mathcal{M} starts in the initial distribution μ_0 .

In the following, we consider the trace equivalence problem of SL-hqMC. As shown in [4], the issue of trace equivalence is closely related to model checking linear-time properties of nonprobabilistic transition systems. For probabilistic systems, this problem was also discussed in [7]. The definition of trace equivalence is as follows.

Definition 8. *Two SL-hqMC \mathcal{M}_1 and \mathcal{M}_2 with the same labeling set Σ are trace equivalent if $P_{\mathcal{M}_1}(w) = P_{\mathcal{M}_2}(w)$ for all $w \in \Sigma^+$.*

In the following, we transform the trace equivalence problem of SL-hqMC to the equivalence problem of finite C-HQA that is decidable in polynomial time as shown in Theorem 2.

Lemma 4. *For every SL-hqMC \mathcal{M} with labeling set Σ , we can construct in linear time a finite C-HQA \mathcal{A} such that $P_{\mathcal{A}}(w) = P_{\mathcal{M}}(w)$ for all $w \in \Sigma^+$.*

Proof. Let $\mathcal{M} = (\mathcal{H}, S, M, \mu_0, AP, L)$ be an SL-hqMC. Note that $\Sigma = 2^{AP}$. We construct C-HQA $\mathcal{A} = (\mathcal{H}, S', \Sigma, \mu'_0, \{M_\sigma\}_{\sigma \in \Sigma}, F)$ as follows.

- $S' = S \cup \{\tau\}$;
- $\mu'_0(s) = \mu_0(s)$ for all $s \in S$ and $\mu'_0(\tau)$ is the zero operator in $\mathcal{P}(\mathcal{H})$;
- $F = S$;
- For each $\sigma \in \Sigma$, M_σ is constructed as

$$M_\sigma = \hat{M}D_\sigma$$

where \hat{M} is a block matrix:

$$\hat{M} = \begin{pmatrix} M & \mathbf{0} \\ \mathbf{0} & \mathcal{I} \end{pmatrix}.$$

That is, $\hat{M}(s, t) = M(s, t)$ for $s, t \in S$, $\hat{M}(\tau, \tau) = \mathcal{I}$, and others are $\mathbf{0}$. D_σ is given by

$$D_\sigma = \begin{pmatrix} \delta_{L(s_1), \sigma} \mathcal{I} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \delta_{L(s_2), \sigma} \mathcal{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \mathbf{0} & \ddots & \mathbf{0} & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \delta_{L(s_n), \sigma} \mathcal{I} & \mathbf{0} \\ \bar{\delta}_{L(s_1), \sigma} \mathcal{I} & \bar{\delta}_{L(s_2), \sigma} \mathcal{I} & \cdots & \bar{\delta}_{L(s_n), \sigma} \mathcal{I} & \mathcal{I} \end{pmatrix}.$$

That is, $D_\sigma(s, s) = \delta_{L(s), \sigma} \mathcal{I}$, $D_\sigma(\tau, s) = \bar{\delta}_{L(s), \sigma} \mathcal{I}$ for all $s \in S$, $D_\sigma(\tau, \tau) = \mathcal{I}$, and others are $\mathbf{0}$. In the above, for $s \in S$ and $\sigma \in \Sigma$, the meaning of $\delta_{L(s), \sigma}$ is

$$\delta_{L(s), \sigma} = \begin{cases} 1, & \text{if } L(s) = \sigma; \\ 0, & \text{otherwise.} \end{cases}$$

In addition, $\bar{\delta}_{L(s), \sigma} = 1 - \delta_{L(s), \sigma}$.

It is obvious that \hat{M} and D_σ satisfy the property that the sum of each column is a trace-preserving quantum operation, and their multiplication also satisfies this property.

By the above construction, \mathcal{A} satisfies the following property.

Proposition 1. *Let $w = \sigma_0 \sigma_1 \cdots \sigma_k \in \Sigma^{k+1}$ and $\mu'_w = M_w \mu'_0$. Then for any $s \in S$, we have*

$$\mu'_w(s) = \sum_{L(s_0 \cdots s_k) = w} M(s, s_k) \prod_{i=1}^k M(s_i, s_{i-1})(\mu'_0(s_0)), \quad (9)$$

where $L(s_0 \cdots s_k) = w$ stands for “ $s_0, \dots, s_k \in S : L(s_0 \cdots s_k) = w$ ” and we will always adopt this succinct notation in the sequel.

Proof. We prove it by induction on k . When $k = 0$, for $s \in S$ we have

$$\begin{aligned}
\mu'_{\sigma_0}(s) &= (M_{\sigma_0}\mu'_0)(s) = (\hat{M}D_{\sigma_0}\mu'_0)(s) \\
&= \sum_{s_0 \in S'} \hat{M}(s, s_0)(D_{\sigma_0}\mu'_0)(s_0) \\
&= \sum_{s_0 \in S} M(s, s_0)(D_{\sigma_0}\mu'_0)(s_0) \\
&= \sum_{s_0 \in S} \delta_{L(s_0)\sigma_0} M(s, s_0)(\mu'_0(s_0)) \\
&= \sum_{L(s_0)=\sigma_0} M(s, s_0)(\mu'_0(s_0)).
\end{aligned}$$

Suppose the result holds for k . Then for $s \in S$ and $\sigma_{k+1} \in \Sigma$ we have

$$\begin{aligned}
\mu'_{w\sigma_{k+1}}(s) &= (M_{\sigma_{k+1}}\mu'_w)(s) = (\hat{M}D_{\sigma_{k+1}}\mu'_w)(s) \\
&= \sum_{s_{k+1} \in S'} \hat{M}(s, s_{k+1})(D_{\sigma_{k+1}}\mu'_w)(s_{k+1}) \\
&= \sum_{s_{k+1} \in S} M(s, s_{k+1})(D_{\sigma_{k+1}}\mu'_w)(s_{k+1}) \\
&= \sum_{s_{k+1} \in S} \delta_{L(s_{k+1})\sigma_{k+1}} M(s, s_{k+1})(\mu'_w(s_{k+1})) \\
&\stackrel{(b)}{=} \sum_{s_{k+1} \in S} \delta_{L(s_{k+1})\sigma_{k+1}} M(s, s_{k+1}) \sum_{L(s_0 \cdots s_k)=w} M(s_{k+1}, s_k) \prod_{i=1}^k M(s_i, s_{i-1})(\mu'_0(s_0)) \\
&= \sum_{L(s_0 \cdots s_{k+1})=w\sigma_{k+1}} M(s, s_{k+1}) \prod_{i=1}^{k+1} M(s_i, s_{i-1})(\mu'_0(s_0))
\end{aligned}$$

where (b) is achieved by substituting Eq. (9) for $\mu'_w(s_{k+1})$. Thus we have proved Proposition 1. \square

Now for $w = \sigma_0 \sigma_1 \cdots \sigma_k \in \Sigma^+$, the accepting probability of \mathcal{A} is

$$\begin{aligned}
P_{\mathcal{A}}(w) &= \text{Tr} \left(\sum_{s \in S} \mu'_w(s) \right) \\
&= \text{Tr} \left(\sum_{s \in S} \sum_{L(s_0 \cdots s_k) = w} M(s, s_k) \prod_{i=1}^k M(s_i, s_{i-1})(\mu'_0(s_0)) \right) \\
&= \sum_{L(s_0 \cdots s_k) = w} \text{Tr} \left(\sum_{s \in S} M(s, s_k) \prod_{i=1}^k M(s_i, s_{i-1})(\mu'_0(s_0)) \right) \\
&\stackrel{(c)}{=} \sum_{L(s_0 \cdots s_k) = w} \text{Tr} \left(\prod_{i=1}^k M(s_i, s_{i-1})(\mu'_0(s_0)) \right) \\
&= P_{\mathcal{M}}(w)
\end{aligned}$$

where (c) holds because $\sum_{s \in S} M(s, t_k)$ is trace-preserving. This completes the proof of Lemma 4. \square

Therefore, based on Theorem 2 and Lemma 4, we obtain the following result.

Theorem 3. *The trace equivalence problem of SL-hqMC is decidable in polynomial time.*

6. Quantitative analysis of linear-time properties

Recall that T^* denotes the set of all finite-length sequence over nonempty set T . We also need another notation T^ω that stands for the set of all infinite-length sequence over T . Let $\mathcal{M} = (\mathcal{H}, S, M, \mu_0, AP, L)$ be an SL-hqMC. A path π of \mathcal{M} is an infinite sequence of states $s_0 s_1 \cdots \in S^\omega$ where $M(s_{i+1}, s_i) \neq \mathbf{0}_{\mathcal{H}}$ for all $i \geq 0$. A finite path $\hat{\pi}$ is a finite prefix of a path. The sets of all infinite and finite paths of \mathcal{M} starting in state s are denoted by $Path^{\mathcal{M}}(s)$ and $Path_{fin}^{\mathcal{M}}(s)$, respectively.

6.1. Super-operator valued measure

It is a central problem to determine the accumulated super-operator along certain paths for reasoning about the behavior of an SL-hqMC. For example, if we can first determine the accumulated super-operator $\prod_{i=1}^k M(s_i, s_{i-1})$ in Eq. (8), then it is easy to compute the state $\rho_{\bar{s}}$ for an arbitrarily given initial state μ_0 . To this end, the *super-operator valued measure* (SVM for short) was proposed in [8], which plays a similar role as probability measure for probabilistic systems. We recall the definition of SVM and some related facts as follows.

Definition 9. Let (Ω, Σ) be a measurable space; that is, Ω is a non-empty set and Σ a σ -algebra over Ω . A function $\Delta : \Sigma \rightarrow \mathcal{SI}(\mathcal{H})$ is said to be a super-operator valued measure (SVM for short) if Δ satisfies the following properties:

1. $\Delta(\Omega) \approx \mathcal{I}$;
2. $\Delta(\biguplus_i A_i) \approx \sum_i \Delta(A_i)$ for all pairwise disjoint and countable sequence A_1, A_2, \dots in Ω .

We call the triple (Ω, Σ, Δ) a (super-operator valued) measure space.

Given an SL-hqMC \mathcal{M} and a state $s \in S$, for any finite path $\hat{\pi} = s_0 \dots s_n \in \text{Path}_{fin}^{\mathcal{M}}(s)$, we define the super-operator

$$\mathbf{Q}(\hat{\pi}) = \begin{cases} \mathcal{I}, & \text{if } n = 0; \\ M(s_n, s_{n-1}) \cdots M(s_1, s_0), & \text{otherwise.} \end{cases}$$

Next we define the cylinder set $\text{Cyl}(\hat{\pi}) \subseteq \text{Path}^{\mathcal{M}}(s)$ as

$$\text{Cyl}(\hat{\pi}) = \{\pi \in \text{Path}^{\mathcal{M}}(s) : \hat{\pi} \text{ is a prefix of } \pi\};$$

that is, the set of all infinite paths with prefix $\hat{\pi}$. Let

$$\mathcal{S}^{\mathcal{M}}(s) = \{\text{Cyl}(\hat{\pi}) : \hat{\pi} \in \text{Path}_{fin}^{\mathcal{M}}(s)\} \cup \{\emptyset\}.$$

Q_s is a mapping from $\mathcal{S}^{\mathcal{M}}(s)$ to $\mathcal{SI}(\mathcal{H})$, defined by letting $Q_s(\emptyset) = \mathbf{0}_{\mathcal{H}}$ and

$$Q_s(\text{Cyl}(\hat{\pi})) = \mathbf{Q}(\hat{\pi}).$$

Then we have the following result.

Lemma 5 ([8]). *The mapping Q_s defined above can be extended to a SVM, denoted by Q_s again, on the σ -algebra generated by $\mathcal{S}^{\mathcal{M}}(s)$. Furthermore, this extension is unique up to the equivalence relation \approx .*

6.2. Linear-time Properties

A linear-time (LT) property over the atomic proposition set AP is defined to be a subset P of $(2^{AP})^{\omega}$. In the remainder of this section, we consider some special classes of linear-time properties. Safety is one of the most important kinds of linear-time properties. A safety property specifies that “something bad never happens”.

Definition 10. *An LT property P over AP is called a safety property if for all words $\sigma \in (2^{AP})^{\omega} \setminus P$ there exists a finite prefix $\hat{\sigma} \in (2^{AP})^*$ of σ such that*

$$P \cap \{\sigma' \in (2^{AP})^{\omega} : \hat{\sigma} \text{ is a finite prefix of } \sigma'\} = \emptyset.$$

Any such finite word $\hat{\sigma}$ is called a *bad prefix* of P . We write $BPref(P)$ for the set of bad prefixes of P . Note that $BPref(P)$ is a language over $\Sigma = 2^{AP}$. A safety property P is called a *regular safety property*, if its bad prefix set $BPref(P)$ is a regular language.

As we know, for each regular language there exists an NFA accepting it. An NFA is a tuple $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ where Q is finite state set, Σ is a finite alphabet, $\delta : Q \times \Sigma \rightarrow 2^Q$ is a transition function, $Q_0 \subseteq Q$ denotes a set of initial states, and $F \subseteq Q$ is called the accepting set. We often write $q \xrightarrow{a} p$ if $p \in \delta(q, a)$ where $q, p \in Q$ and $a \in \Sigma$. A string $w = a_1 a_2 \cdots a_n \in \Sigma^*$ is said to be *accepted* by \mathcal{A} , if there exists a finite state sequence $q_0 q_1 \cdots q_n$ such that $q_0 \in Q_0$, $q_{i-1} \xrightarrow{a_i} q_i$ for $1 \leq i \leq n$, and $q_n \in F$. The language accepted by \mathcal{A} , denoted by $\mathcal{L}(\mathcal{A})$, is the set of strings over Σ that are accepted by \mathcal{A} .

A DFA is a special NFA $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ where $|Q_0| = 1$ and $|\delta(q, a)| \leq 1$ for all $q \in Q$ and $a \in \Sigma$. Then a DFA is usually denoted by $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ where q_0 is the unique initial state. If it is further required that $|\delta(q, a)| = 1$ for all $q \in Q$ and $a \in \Sigma$, then \mathcal{A} is called a *total* DFA. Note that NFA, DFA and total DFA accept the same language class, i.e., regular languages. In the sequel, when mentioning a DFA, we always assume it is total.

Therefore, a safety property can be characterized by a DFA.

6.3. Quantitative analysis of regular safety properties

Given an SL-qhMC $\mathcal{M} = (\mathcal{H}, S, M, \mu_0, AP, L)$, a state $s \in S$, and an LT property P , let

$$Q_s(s \models P) = Q_s\{\pi \in Path^{\mathcal{M}}(s) : L(\pi) \in P\}$$

where $L(\pi) = L(s_0)L(s_1)\cdots$ is called the *trace* of path $\pi = s_0 s_1 \cdots$. In the following we show how to determine this quantity if P is a regular safety property. First we note that $Q_s(s \models P) + Q_s(s \not\models P) \approx \mathcal{I}$ where

$$\begin{aligned} Q_s(s \not\models P) &= Q_s\{\pi \in Path^{\mathcal{M}}(s) : L(\pi) \notin P\} \\ &= Q_s\{\pi \in Path^{\mathcal{M}}(s) : pref(L(\pi)) \cap \mathcal{L}(\mathcal{A}) \neq \emptyset\} \end{aligned}$$

where $pref(A_0 A_1 \cdots)$ denotes the set of all finite prefixes of $A_0 A_1 \cdots \in (2^{AP})^\omega$, and \mathcal{A} is a DFA accepting $BPref(P)$. In order to get the quantity $Q_s(s \not\models P)$, we need the concept of product between SL-hqMC and DFA.

Definition 11. Let $\mathcal{M} = (\mathcal{H}, S, M, \mu_0, AP, L)$ be an SL-hqMC and $\mathcal{A} = (Q, 2^{AP}, \delta, q_0, F)$ be a DFA. Then product $\mathcal{M} \otimes \mathcal{A}$ is the SL-hqMC:

$$\mathcal{M} \otimes \mathcal{A} = (\mathcal{H}, S \times Q, M', \mu'_0, \{accept\}, L')$$

where $L'(\langle s, q \rangle) = \{\text{accept}\}$ if $q \in F$ and $L'(\langle s, q \rangle) = \emptyset$ otherwise, and

$$\mu'_0(\langle s, q \rangle) = \begin{cases} \mu_0(s), & \text{if } q = \delta(q_0, L(s)); \\ \mathbf{0}, & \text{otherwise.} \end{cases}$$

The transition mapping in $\mathcal{M} \otimes \mathcal{A}$ is given by

$$M'(\langle s', q' \rangle, \langle s, q \rangle) = \begin{cases} M(s', s), & \text{if } q' = \delta(q, L(s')); \\ \mathbf{0}, & \text{otherwise.} \end{cases}$$

For each path $\pi = s_0 s_1 s_2 \dots$ in \mathcal{M} , there exists a unique sequence of states $q_0 q_1 q_2 \dots$ in \mathcal{A} for $L(\pi) = L(s_0)L(s_1)L(s_2)\dots$ such that

$$q_0 \xrightarrow{L(s_0)} q_1 \xrightarrow{L(s_1)} q_2 \xrightarrow{L(s_2)} \dots$$

and

$$\pi^+ = \langle s_0, q_1 \rangle \langle s_1, q_2 \rangle \langle s_2, q_3 \rangle \dots$$

is a path in $\mathcal{M} \otimes \mathcal{A}$. Vice versa, every path in $\mathcal{M} \otimes \mathcal{A}$ which starts in state $\langle s, \delta(q_0, L(s)) \rangle$ arises from the combination of a path in \mathcal{M} and a corresponding state sequence in \mathcal{A} . The corresponding relation between the states are depicted in Fig. 3.

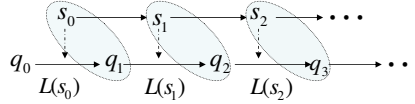


Figure 3: States in $\mathcal{M} \otimes \mathcal{A}$.

Note that the DFA \mathcal{A} does not affect the accumulated super-operator along a path. That is, for each measurable set $\Pi \subseteq \text{Path}^{\mathcal{M}}(s)$,

$$Q_s^{\mathcal{M}}(\Pi) = Q_{\langle s, \delta(q_0, L(s)) \rangle}^{\mathcal{M} \otimes \mathcal{A}} \underbrace{\{\pi^+ : \pi \in \Pi\}}_{\Pi^+}$$

where the superscripts \mathcal{M} and $\mathcal{M} \otimes \mathcal{A}$ are used to indicate the underlying systems. In particular, if Π is the set of paths that start in s and refute regular safety property P , i.e.,

$$\Pi = \{\pi \in \text{Path}^{\mathcal{M}}(s) : \text{pref}(L(\pi)) \cap \mathcal{L}(\mathcal{A}) \neq \emptyset\}$$

where \mathcal{A} is a DFA accepting $B\text{Pref}(P)$, then Π^+ is the set of paths in $\mathcal{M} \otimes \mathcal{A}$ that start in $\langle s, \delta(q_0, L(s)) \rangle$ and eventually reach an accept state of \mathcal{A} :

$$\Pi^+ = \{\pi^+ \in \text{Path}^{\mathcal{M} \otimes \mathcal{A}}(\langle s, \delta(q_0, L(s)) \rangle) : L'(\pi^+) \in \Diamond \text{accept}\}.$$

Here the linear temporal logic notation “ $\diamond accept$ ” is used to denote the LT property over $AP = \{accept\}$ consisting of sequence $A_0A_1 \cdots \in (2^{AP})^\omega$ for which there exists a finite index i such that $A_i = \{accept\}$. This shows that $Q_s(s \not\models P)$ can be derived from the accumulated super-operator for $\mathcal{M} \otimes \mathcal{A}$ reaching the set $B = S \times F$ starting from $\langle s, \delta(q_0, L(s)) \rangle$. The latter problem, known as the *reachability problem*, can be solved by using Theorem 2.5 in [8]. This is formally stated as follows.

Theorem 4. *Let P be a regular safety property, \mathcal{A} a DFA for the set of bad prefixes P , \mathcal{M} an SL-hqMC, and s a state of \mathcal{M} . Then:*

$$\begin{aligned} Q_s(s \models P) &\approx \mathcal{I} - Q_s(s \not\models P) \\ &\approx \mathcal{I} - Q_{\langle s, q_s \rangle}^{\mathcal{M} \otimes \mathcal{A}}(\langle s, q_s \rangle \models \diamond accept) \end{aligned}$$

where $q_s = \delta(q_0, L(s))$.

Remark 4. Note that in the above procedure, we used an assumption that $\epsilon \notin \mathcal{L}(A)$. Otherwise, the proof would fail. However, this is not a severe restriction since if $\epsilon \in \mathcal{L}(A)$, then all finite words over 2^{AP} are bad prefixes, and hence, $P = \emptyset$. In this case, $Q_s(s \models P) = \mathbf{0}$.

6.4. Questions on quantitative analysis of ω -regular properties

In the above, we have taken a quantitative analysis of regular safety properties. Then it is natural to consider the quantitative analysis of more general LT properties; for example, how about ω -regular properties? ω -regular properties are a much larger family of LT properties than regular safety properties, and they are characterized by Büchi automata.

A *nondeterministic Büchi automaton* (NBA) is represented by the same tuple $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ as an NFA, but with a different accepting condition. An infinite string $w = a_1a_2 \cdots \in \Sigma^\omega$ is said to be accepted by NBA \mathcal{A} , if there exists a sequence $q_0q_1 \cdots \in Q^\omega$ such that $q_0 \in Q_0$, $q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \cdots$, and $q_i \in F$ for infinitely many indexes $i \geq 0$. The language accepted by NBA \mathcal{A} , denoted by, $\mathcal{L}_\omega(\mathcal{A})$, is the set of all infinite strings over Σ that are accepted by \mathcal{A} . The class of languages accepted by NBA are called *ω -regular languages*. An NBA is called a *deterministic Büchi automaton* (DBA) if its underlying automaton is a DFA. Note that DBA can only accept a proper subset of ω -regular languages.

An LT property P over AP is called a *ω -regular property*, if P is a ω -regular language over the alphabet $\Sigma = 2^{AP}$. If P is a ω -regular property, then its complement $(2^{AP})^\omega \setminus P$ is also a ω -regular property. Note that the regular

safety property discussed before is a special case of ω -regular properties. The reader can refer to [4] for the details about ω -regular properties.

Now we consider the problem of quantitative analysis of ω -regular properties. Formally, given a state s of SL-hqMC \mathcal{M} and a ω -regular property P , how to determine the following value:

$$Q_s(s \models P) = Q_s\{\pi \in Path^{\mathcal{M}}(s) : L(\pi) \in P\}.$$

Here we consider a relatively simple case, that is, assume $P = \mathcal{L}_\omega(\mathcal{A})$ for a DBA \mathcal{A} . Then by similar technical treatments as before, we get that

$$Q_s(s \models P) = Q_{\langle s, q_s \rangle}^{\mathcal{M} \otimes \mathcal{A}}(\langle s, q_s \rangle \models \Box \Diamond accept)$$

where $q_s = \delta(q_0, L(s))$, and “ $\Box \Diamond accept$ ” denotes the LT property over $AP = \{accept\}$ consisting of sequence $A_0 A_1 \cdots \in (2^{AP})^\omega$ for which there exist infinite many indexes i such that $A_i = \{accept\}$. Intuitively, it means that the product system $\mathcal{M} \otimes \mathcal{A}$ starts in $\langle s, q_s \rangle$ and visits the set $S \times F$ infinitely often.

At first glance, one may think this quantity can be determined as done for probabilistic systems [4]. It is, however, not so easy as it looks like. In the probabilistic case, \mathcal{M} is a Markov chain and the SVM measure Q_s is replaced by the probability measure P_s . Then, calculating $P_s(s \models P)$ is finally reduced to finding *bottom strongly connected components* (SCCs that once entered cannot be left anymore) in the underlying graph (a digraph obtained by erasing the edge labels from the transition graph) of Markov chain $\mathcal{M} \otimes \mathcal{A}$. The latter problem depends only on the topological structure and has no relation with the actual transition probability, and thus can be solved by using graph-theoretical searching algorithms. However, this method no longer takes effect in quantum cases, since two states connected in the underlying graph are not necessarily connected in the corresponding quantum Markov chain. In order to explain this point more clearly, we have a look at an example below.

In Fig. 4 the two systems have the same underlying graph. It can be seen that the reachability in MC is consistent with that in its underlying graph. For example, s_3 is reachable from s_0 by passing s_2 in the underlying graph, and at the same time s_3 can be reached from s_0 with probability $\frac{1}{4}$ in MC. However, this consistency no longer holds for hqMC. For instance, s_3 is not reachable from s_0 in qMC, since the accumulated super-operator along $s_0 s_2 s_3$ is $\mathbf{0}$. Therefore, the quantitative analysis of general ω -regular properties for quantum Markov chains cannot be addressed by using graph-theoretical algorithms. Whether this problem is solvable or not is currently not clear and needs further exploration.

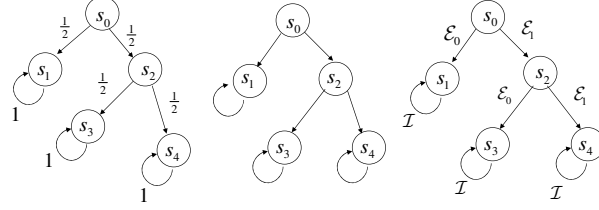


Figure 4: Difference between MC(left) and hqMC (right): Both are given by their transition graphs and the middle is their underlying graph that is obtained by erasing the edge labels. The underlying space of hqMC is a qubit and \mathcal{E}_0 and \mathcal{E}_1 have operation element sets $\{|0\rangle\langle 0|\}$ and $\{|1\rangle\langle 1|\}$, respectively.

7. Conclusion

In this paper, we have studied a novel model of quantum Markov chains. Although this model has the same expressive power as the conventional one, we have shown that it is very suited to describe hybrid systems that consist of a quantum component and a classical one. Based on the quantum Markov chain model, we have further proposed an automaton model called hybrid quantum automata that can be used to describe hybrid systems that receive input (or actions) from the outer world. The language equivalence problem of hybrid quantum automata has been shown to be decidable in polynomial time, and furthermore we have applied this result to the trace equivalence problem of quantum Markov chains which is thus also decidable in polynomial time. Finally, we have discussed model checking linear-time properties of hybrid systems, showing that the quantitative analysis of regular safety properties can be addressed successfully as done for stochastic systems, but the problem for general ω -regular properties is more difficult and needs further exploration.

Hybrid systems modeled by quantum Markov chains have already been often encountered in quantum information processing, and the quantum engineering systems developed in the future will most probably be hybrid systems. Therefore, it is worth developing a theory for describing and verifying these hybrid systems. We hope this work can stimulate further discussion.

Acknowledgement

The authors are thankful to the anonymous referee for valuable comments and suggestions, and to Professor Mingsheng Ying for carefully reading the

manuscript and giving some helpful suggestions. Especially, the first author is greatly indebted to Professor Ying for his guidance when the first author was working in QCIS of UTS. This work is supported in part by the National Natural Science Foundation of China (Nos. 61100001, 61472452, 61272058, 61428208 and 61472412), the National Natural Science Foundation of Guangdong province of China (No. 2014A030313157), Australian Research Council under Grant DP130102764, and the CAS/SAFEA International Partnership Program for Creative Research Team.

References

- [1] D. Aharonov, A. Ambainis, J. Kempe, and U. Vazirani, Quantum walks on graphs, in Proceedings of the 33rd ACM Symposium on Theory of Computing, ACM, 2001, pp. 50-59.
- [2] L. Accardi and D. Koroliuk, Stopping Times for Quantum Markov Chains, *Journal of Theoretical Probability* 5 (1992) 251-535.
- [3] A. Ambainis, J. Watrous, Two-way finite automata with quantum and classical states, *Theoret. Comput. Sci.*287 (2002) 299-311.
- [4] C. Baier and J.-P. Katoen, Principles of Model Checking, the MIT Press, 2008.
- [5] A. Bertoni, C. Mereghetti, and B. Palano, Quantum Computing: 1-Way Quantum Automata, in Proceedings of the 9th International Conference on Developments in Language Theory , LNCS, vol. 2710, Springer, 2003, pp.1-20.
- [6] L. Christoff, I. Christoff, Efficient algorithms for verification of equivalences for probabilistic processes, CAV1991. LNCS, vol. 575, Springer, 1992, pp. 310-321.
- [7] L. Doyen, T. A. Henzinger, J.-F. Raskin, Equivalence of labeled Markov chains. *Int. J. Found. Comput. Sci.* 19(3) (2008) 549-563.
- [8] Y. Feng, N. K. Yu, M. S. Ying, Model checking quantum Markov chains, *J. Comput. System Sci.* 79(2013) 1181-1198.
- [9] S. Gudder. Quantum Markov chains. *Journal of Mathematical Physics*, 49(7) (2008) 072105.

- [10] T. A. Henzinger, The theory of hybrid automata, LICS96, 1996, pp. 278- 292.
- [11] M. Hirvensalo, Quantum Automata with Open Time Evolution, International Journal of Natural Computing Research, 1(2010) 70-85.
- [12] S. Kiefer, A. S. Murawski, J. Ouaknine, B. Wachter, and J. Worrell, APEX: An Analyzer for Open Probabilistic Programs, CAV 2012, LNCS, vol. 7358, Springer, 2012, pp. 693-698.
- [13] S. Kiefer, A. S. Murawski, J. Ouaknine, B. Wachter, and J. Worrell, Language Equivalence for Probabilistic Automata, CAV2011, LNCS vol. 6806, Springer, 2011, pp. 526-540.
- [14] L. Z. Li and Y. Feng, On hybrid models of quantum finite automata, J. Comput. System Sci. 81(2015) 1144-1158.
- [15] L. Z. Li and D. W. Qiu et al, Characterizations of one-way general quantum finite automata, Theoret. Comput. Sci. 419 (2012) 73-91.
- [16] L. Z. Li and D. W. Qiu, Determining the equivalence for one-way quantum finite automata, Theoret. Comput. Sci. 403 (2008) 42-51.
- [17] C. Liu and N. Petulante, On Limiting Distributions of Quantum Markov Chains, International Journal of Mathematics and Mathematical Sciences, Volume 2011, Article ID 740816, 2011.
- [18] A. S. Murawski and J. Ouaknine, On Probabilistic Program Equivalence and Refinement, CONCUR2005, LNCS vol. 3653, Springer, 2005, pp. 156-170.
- [19] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, Cambridge, 2000.
- [20] A. Paz, Introduction to Probabilistic Automata, Academic Press, New York, 1971.
- [21] D. W. Qiu, L. Z. Li, P. Mateus and A. Sernadas, Exponentially more concise quantum recognition of non-RMM regular languages, J. Comput. System Sci. 81 (2015) 359-375.
- [22] D.W. Qiu, L.Z. Li, P. Mateus, J. Gruska, Quantum finite automata, in: Finite State Based Models and Applications (Edited by Jiacun Wang), CRC Handbook, 2012.

- [23] J. F. Raskin, An Introduction to Hybrid Automata, Handbook of Networked and Embedded Control Systems Control Engineering, IV, 491-517, Springer, 2005.
- [24] P. Selinger. Towards a quantum programming language. *Mathematical Structures in Computer Science* 14 (4) (2004) 527-586.
- [25] W.G. Tzeng, A Polynomial-time Algorithm for the Equivalence of Probabilistic Automata, *SIAM J. Comput.* 21 (2) (1992) 216-227.
- [26] J. Watrous, On the complexity of simulating space-bounded quantum computations, *Computational Complexity* 12 (2003) 48-84.
- [27] J. Watrous, Lecture notes on theory of quantum information, <http://www.cs.uwaterloo.ca/~watrous>.
- [28] M. S. Ying, N. K. Yu, Y. Feng, and R. Y. Duan, Verification of Quantum Programs, *Science of Computer Programming* 78 (9) (2013) 1679-1700.
- [29] S. G. Zheng, D. W. Qiu, J. Gruska, Power of the interactive proof systems with verifiers modeled by semi-quantum two-way finite automata, *Inform. Comput.*, 241 (2015) 197-214.
- [30] S. G. Zheng, D. W. Qiu, L. Z. Li and J. Gruska, One-way finite automata with quantum and classical states, *Dassow Festschrift 2012*, LNCS, vol. 7300, Springer, 2012, pp. 273-290.
- [31] S.G. Zheng, J. Gruska, and D. W. Qiu, On the State Complexity of Semi-quantum Finite Automata, *RAIRO-Theoretical Informatics and Applications* 48 (2014) 187-207.