

Model Checking Quantum Continuous-Time Markov Chains

Ming Xu  

Shanghai Key Lab of Trustworthy Computing,
MoE Engineering Research Center of Software/Hardware Co-design Technology and Application,
East China Normal University, Shanghai, China

Jingyi Mei  

Shanghai Key Lab of Trustworthy Computing, East China Normal University, Shanghai, China

Ji Guan  

State Key Lab of Computer Science, Institute of Software,
Chinese Academy of Sciences, Beijing, China

Nengkun Yu  

Centre for Quantum Software and Information, Faculty of Engineering and Information Technology,
University of Technology, Sydney, Australia

Abstract

Verifying quantum systems has attracted a lot of interests in the last decades. In this paper, we initialise the model checking of quantum continuous-time Markov chain (QCTMC). As a real-time system, we specify the temporal properties on QCTMC by signal temporal logic (STL). To effectively check the atomic propositions in STL, we develop a state-of-the-art real root isolation algorithm under Schanuel's conjecture; further, we check the general STL formula by interval operations with a bottom-up fashion, whose query complexity turns out to be linear in the size of the input formula by calling the real root isolation algorithm. A running example of an open quantum walk is provided to demonstrate our method.

2012 ACM Subject Classification Theory of computation → Verification by model checking; Computing methodologies → Symbolic and algebraic algorithms; Theory of computation → Quantum computation theory

Keywords and phrases Model Checking, Formal Logic, Quantum Computing, Computer Algebra

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2021.13

Related Version *Previous Version:* <https://arxiv.org/abs/2105.00382>

Funding *Ming Xu:* The corresponding author M. X. is supported by the National Natural Science Foundation of China (No. 11871221), the National Key R&D Program of China (No. 2018YFA0306704), the Fundamental Research Funds for the Central Universities, and the Research Funds of Happiness Flower ECNU (No. 2020ECNU-XFZH005).

Ji Guan: The corresponding author J. G. is supported by the National Key R&D Program of China (No. 2018YFA0306701) and the National Natural Science Foundation of China (No. 61832015).

Nengkun Yu: N. Y. is supported by ARC Discovery Early Career Researcher Award (No. DE180100156) and ARC Discovery Program (No. DP210102449).

1 Introduction

Aiming to study nature, physicists use different mechanics depending on the scale of the objects they are interested in. Classical mechanics describes nature at a macroscopic scale (far larger than 10^{-9} meters), while quantum mechanics is applied at a microscopic scale (near or less than 10^{-9} meters).



© Ming Xu, Jingyi Mei, Ji Guan, and Nengkun Yu;
licensed under Creative Commons License CC-BY 4.0

32nd International Conference on Concurrency Theory (CONCUR 2021).

Editors: Serge Haddad and Daniele Varacca; Article No. 13; pp. 13:1–13:17

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

13:2 Model Checking Quantum CTMCs

In physics, a *closed* quantum system is a physical system that does not interchange information (i.e. energy and/or matter) with another system. The behavior of a closed quantum system can be described as follows: A particle at this level can be mathematically represented by a normalised complex vector $|s\rangle$ in a Hilbert space \mathcal{H} . The time evolution of a single particle *closed* system is described by the Schrödinger equation:

$$\frac{d|s(t)\rangle}{dt} = -i\mathbf{H}|s(t)\rangle \quad (1)$$

with some *Hamiltonian* \mathbf{H} (a Hermitian matrix on \mathcal{H}), where $|s(t)\rangle$ is the state of the system at time t .

More practically, an *open* quantum system interacting with the surrounding environment need to be considered. An open quantum system is a quantum-mechanical system that interacts with an external quantum system, which is known as the environment or a bath. In general, these interactions significantly change the dynamics of the system and result in quantum dissipation, such that the information contained in the system is lost to its environment. Suffering noises from the environment, the state of the system cannot be completely known. Thus a *density operator* ρ (positive semidefinite matrix with unit trace) on \mathcal{H} is introduced to describe the uncertainty of the possible states: $\rho = \sum_{i \in I} p_i |s_i\rangle\langle s_i|$, where $\{(p_i, |s_i\rangle)\}_{i \in I}$ is a mixed state or an ensemble expressing that the quantum state is at $|s_i\rangle$ with probability p_i , and $\langle s_i|$ is the complex conjugate and transpose of $|s_i\rangle$. In this case, the evolution is described by the Lindblad's master equation:

$$\frac{d\rho(t)}{dt} = \mathcal{L}(\rho(t)) \quad (2)$$

where $\rho(t)$ stands for the (possibly mixed) state of the system, and \mathcal{L} is a linear function of $\rho(t)$ (to be formally described in Subsection 2.2), which is generally irreversible.

To reveal physical phenomenon, physicists have intensively studied the properties of closed and open quantum systems case by case in the last decades, such as long-term behaviors (e.g. [18]) and stabilities (e.g. [10]). In recent years, the computer science community has stepped into this field and adopted model checking technique to study quantum systems [16, 17]. Specifically, the quantum systems can be simulated by some mathematical models and a bulk of physical properties can be reformulated as formulas in some temporal logic with atomic propositions of quantum interpretation. In particular, a *quantum discrete-time Markov chain* (QDTMC) $(\mathcal{H}, \mathcal{E})$ has been introduced as quantum generalisations of classical discrete-time Markov chains (DTMC) to model the evolution in Eq. (2) in a single time unit:

$$\rho(t+1) = \mathcal{E}(\rho(t)) \quad (3)$$

where \mathcal{E} is a discretised quantum operation obtained from the Lindblad's master equation, usually called a *super-operator* in the field of quantum information and computation. Several fundamental model checking-related problems for QDTMCs have been studied in the literature, including limiting states [40, 21], reachability [45, 41], repeated reachability [14], linear time properties [30], and persistence based on irreducible and periodic decomposition techniques [8, 21]. These techniques were equipped to solve real-world problems in several different areas. For example, [15] proposed algorithms to model check quantum cryptographic protocols against a quantum extension of probabilistic computation tree logic (PCTL); and linear temporal logic (LTL) was adopted to specify a bulk of properties of quantum many-body and statistical systems, and corresponding model checking algorithm

was developed [20]. For quantum concurrent systems, the reachability and the termination are studied in [47]. Very recently, Yu investigated a quantum temporal logic for the specification of the system and studied the model checking problem for basic formulas [46]. See [43, 44] for the comprehensive review of this research line.

However, to the best of our knowledge, there is no work on model checking the quantum continuous-time system of Eq. (2). In contrast, there are fruitful results in the classical counterpart, which is usually modelled by a continuous-time Markov chain (CTMC).

The seminal work on verifying CTMCs is Aziz *et al.*'s paper [3, 4]. The authors introduced continuous stochastic logic (CSL) interpreted on CTMCs. Roughly speaking, the syntax of CSL amounts to that of PCTL plus the multiphase until formula $\Phi_1 U^{\mathcal{I}_1} \Phi_2 \dots U^{\mathcal{I}_K} \Phi_{K+1}$, for some $K \geq 1$. Because [3] restricts probabilities in $\text{Pr}_{>c}$ to $c \in \mathbb{Q}$, they can show decidability of model checking for CSL using number-theoretic analysis. An approximate model checking algorithm for a reduced version of CSL was provided by Baier *et al.* [6], who restrict path formulas to binary until: $\Phi_1 U^{\mathcal{I}} \Phi_2$. Under this logic, they successfully applied efficient numerical techniques for *transient analysis* [5] using *uniformisation* [37]. The approximate algorithms have been extended for multiphase until formulas using *stratification* [48, 49]. Xu *et al.* considered the multi-phase until formulas over the CTMC with rewards [42]. An integral-style algorithm was proposed to attack this problem, whose effectiveness is ensured by number-theoretic results and algebraic methods. Recently, continuous linear logic was introduced to specify on CTMCs, whose decidability was established [23]. Most the above algorithms have been implemented in probabilistic model checkers, like *PRISM* [29], *Storm* [13], and *ePMC* [24].

Unfortunately, these results from CTMCs cannot directly tackle the problem of automatically verifying quantum continuous-time systems. The main obstacle is that the state space of classical case is finite, while the space of quantum states in even a finite-dimensional Hilbert space is a continuum. Even though the probability space of a CTMC is also a continuum, a probability distribution can only represent one ensemble (i.e. the uncertainty of the possible states), while a quantum state generally admits infinitely many ensembles, such as $\frac{1}{2} |0\rangle\langle 0| + \frac{1}{2} |1\rangle\langle 1| = \frac{1}{2} |\psi_1\rangle\langle \psi_1| + \frac{1}{2} |\psi_2\rangle\langle \psi_2|$ for any two orthogonal normalised complex vectors $|\psi_1\rangle$ and $|\psi_2\rangle$ on the Hilbert space linearly spanned by orthonormal basis $\{|0\rangle, |1\rangle\}$. In this paper, we cross the difficulty by introducing quantum continuous-time Markov chains (QCTMC) to model the evolution of quantum continuous-time systems in Eq. (2) and converting them into a distribution transformer that preserves the laws of quantum mechanics. Then, we consider a wide logic, signal temporal logic (STL), to specify real-time properties against QCTMC. The STL is more expressible than LTL and CTL. Finally we present an exact method to decide the STL formula using real root isolation and interval operations, whose query complexity turns out to be linear in the size of the input formula by calling the real root isolation routine.

The key contributions of the present paper are three-fold:

1. In the field of formal verification, the model checking on DTMC, QDTMC and CTMC has been well studied in the past decades, but no work on checking QCTMCs as far as we know. The first contribution is filling this blank.
2. In order to solve the atomic propositions in STL, we develop a state-of-the-art real root isolation algorithm for a rich class of real-valued functions based on Schanuel's conjecture.
3. We provide a running example – open quantum walk equipped with an absorbing boundary, which drops the restriction on the underlying graph being symmetric/Hermitian. The non-Hermitian structure brings real technical hardness. Fortunately, it is overcome by Eq. (2) employed for describing the dynamical system of QCTMC.

Organisation. The rest of the paper is structured as follows. Section 2 reviews some notions and notations from quantum computing, the Lindblad's master equation and number theory. In Sections 3 and 4, we introduce the model of quantum continuous-time Markov chains and the signal temporal logic (STL), respectively. We solve the atomic propositions in STL in Section 5, and decide the general STL formulas in Section 6. Section 7 is the conclusion.

2 Preliminaries

2.1 Quantum Computing

Let \mathcal{H} be a Hilbert space with dimension d . We employ the Dirac notations that are standard in quantum computing:

- $|s\rangle$ stands for a unit column vector in \mathcal{H} labelled with s ;
- $\langle s| := |s\rangle^\dagger$ is the Hermitian adjoint (complex conjugate and transpose) of $|s\rangle$;
- $\langle s_1|s_2\rangle := \langle s_1||s_2\rangle$ is the inner product of $|s_1\rangle$ and $|s_2\rangle$;
- $|s_1\rangle\langle s_2| := |s_1\rangle \otimes \langle s_2|$ is the outer product, where \otimes denotes tensor product; and
- $|s_1, s_2\rangle := |s_1\rangle|s_2\rangle$ is a shorthand of the product state $|s_1\rangle \otimes |s_2\rangle$.

A linear operator γ is *Hermitian* if $\gamma = \gamma^\dagger$; and it is *positive* if $\langle s|\gamma|s\rangle \geq 0$ holds for all $|s\rangle \in \mathcal{H}$. A *projector* \mathbf{P} is a positive operator of the form $\sum_{i=1}^m |s_i\rangle\langle s_i|$ with $m \leq d$, where $|s_i\rangle$ are orthonormal. Clearly, there is a bijective map between projectors $\mathbf{P} = \sum_{i=1}^m |s_i\rangle\langle s_i|$ and subspaces of \mathcal{H} that are spanned by $\{|s_i\rangle : 1 \leq i \leq m\}$. In sum, positive operators are Hermitian ones whose eigenvalues are nonnegative; and projectors are positive operators whose eigenvalues are 0 or 1. Besides, a linear operator \mathbf{U} is *unitary* if $\mathbf{U}\mathbf{U}^\dagger = \mathbf{U}^\dagger\mathbf{U} = \mathbf{I}$ where \mathbf{I} is the *identity* operator.

The *trace* of a linear operator γ is defined as $\text{tr}(\gamma) := \sum_{i=1}^d \langle s_i|\gamma|s_i\rangle$ for any orthonormal basis $\{|s_i\rangle : 1 \leq i \leq d\}$ of \mathcal{H} . A *density operator* ρ on \mathcal{H} is a positive operator with unit trace. It gives rise to a generic way to describe quantum states: if a density operator ρ is $|s\rangle\langle s|$ for some $|s\rangle \in \mathcal{H}$, ρ is said to be a *pure* state; otherwise it is a *mixed* one, i.e. $\rho = \sum_{i=1}^m p_i |s_i\rangle\langle s_i|$ with $m \geq 2$ by spectral decomposition, where p_i are positive eigenvalues (interpreted as the *probabilities* of taking the pure states $|s_i\rangle$) and their sum is 1. In other words, a pure state indicates the system state which we completely know; a mixed state gives all possible system states, with total probability 1, which we know. We denote by $\mathcal{D}_{\mathcal{H}}$ the set of density operators on \mathcal{H} . The subscript \mathcal{H} of $\mathcal{D}_{\mathcal{H}}$ will be omitted if it is clear from the context.

The system evolution between pure states is characterised by some unitary operator \mathbf{U} , i.e. $|s(t)\rangle\langle s(t)| = \mathbf{U}(t)|s(0)\rangle\langle s(0)|\mathbf{U}^\dagger(t)$ where $\mathbf{U}(t)$ comes from $\exp(-i\mathbf{H}t)$ for the Hermitian operator \mathbf{H} in Eq. (1); the system evolution between density operators (pure or mixed states) is characterised by some completely positive operator-sum, a.k.a. *super-operator*, i.e. $\rho(t) = \sum_{j=1}^m \mathbf{L}_j(t)\rho(0)\mathbf{L}_j^\dagger(t)$ where \mathbf{L}_j are linear operators satisfying the trace preservation $\sum_{j=1}^m \mathbf{L}_j^\dagger\mathbf{L}_j = \mathbf{I}$. The latter dynamical system is obtained in such a way: an enlarged unitary operator acts on the purified composite state $(\sum_{i=1}^m \sqrt{p_i}|s_i, env_i\rangle)(\sum_{i=1}^m \sqrt{p_i}\langle s_i, env_i|)$ with $\rho(0) = \sum_{i=1}^m p_i |s_i\rangle\langle s_i|$ for some orthonormal environment states $|env_i\rangle$ [34, Section 2.5], then the environment in the resulting composite state is traced out, which turns out to be the aforementioned operator-sum form. We call it an *open* system as it interacts with the environment. Whereas, the former dynamical system, independent from the environment, is called a *closed* system. Open systems are more common than closed systems in practice.

2.2 Lindblad's Master Equation

To characterise state evolution of the continuous-time open system with the *memoryless* property, we employ the Lindblad's master equation [31, 19] that is

$$\rho' = \mathcal{L}(\rho) = -i\mathbf{H}\rho + i\rho\mathbf{H} + \sum_{j=1}^m \left(\mathbf{L}_j \rho \mathbf{L}_j^\dagger - \frac{1}{2} \mathbf{L}_j^\dagger \mathbf{L}_j \rho - \frac{1}{2} \rho \mathbf{L}_j^\dagger \mathbf{L}_j \right), \quad (4)$$

where \mathbf{H} is a Hermitian operator and \mathbf{L}_j are linear operators. The terms $-i\mathbf{H}\rho + i\rho\mathbf{H}$ describe the evolution of the internal system; the terms $\mathbf{L}_j \rho \mathbf{L}_j^\dagger - \frac{1}{2} \mathbf{L}_j^\dagger \mathbf{L}_j \rho - \frac{1}{2} \rho \mathbf{L}_j^\dagger \mathbf{L}_j$ describe the interaction between system and environment. In other words, to characterise the evolution of an open system, it is necessary to use those linear operators \mathbf{L}_j besides the Hermitian operator \mathbf{H} . It is known to be the most general type of Markovian and time-homogeneous master equation describing (in general non-unitary) evolution of the system state that preserves the laws of quantum mechanics (i.e., completely positive and trace-preserving for any initial condition).

In the following, we will derive the solution of Eq. (4). We first define two useful functions:

- $\text{L2V}(\gamma) := \sum_{i=1}^d \sum_{j=1}^d \langle i | \gamma | j \rangle |i, j\rangle$ that rearranges entries of the linear operator γ on the Hilbert space \mathcal{H} with dimension d as a column vector; and
- $\text{V2L}(\mathbf{v}) := \sum_{i=1}^d \sum_{j=1}^d \langle i, j | \mathbf{v} | i \rangle \langle j |$ that rearranges entries of the column vector \mathbf{v} as a linear operator.

Here, L2V and V2L are read as “linear operator to vector” and “vector to linear operator”, respectively. They are mutually inverse functions, so that if a linear operator (resp. its vectorisation) is determined, its vectorisation (resp. the original linear operator) is determined. Using the fact that $\mathbf{D} = \mathbf{ABC} \iff \text{L2V}(\mathbf{D}) = (\mathbf{A} \otimes \mathbf{C}^T) \text{L2V}(\mathbf{B})$ holds for any linear operators $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$, we can reformulate Eq. (4) as the linear ordinary differential equation

$$\begin{aligned} \text{L2V}(\rho') &= \left[-i\mathbf{H} \otimes \mathbf{I} + i\mathbf{I} \otimes \mathbf{H}^T + \sum_{j=1}^m \left(\mathbf{L}_j \otimes \mathbf{L}_j^* - \frac{1}{2} \mathbf{L}_j^\dagger \mathbf{L}_j \otimes \mathbf{I} - \frac{1}{2} \mathbf{I} \otimes \mathbf{L}_j^T \mathbf{L}_j^* \right) \right] \text{L2V}(\rho) \\ &= \mathbf{M} \cdot \text{L2V}(\rho), \end{aligned} \quad (5)$$

where $*$ denotes entry-wise complex conjugate. We call $\mathbf{M} = -i\mathbf{H} \otimes \mathbf{I} + i\mathbf{I} \otimes \mathbf{H}^T + \sum_{j=1}^m (\mathbf{L}_j \otimes \mathbf{L}_j^* - \frac{1}{2} \mathbf{L}_j^\dagger \mathbf{L}_j \otimes \mathbf{I} - \frac{1}{2} \mathbf{I} \otimes \mathbf{L}_j^T \mathbf{L}_j^*)$ the *governing matrix* of Eq. (5). As a result, we get the desired solution $\text{L2V}(\rho(t)) = \exp(\mathbf{M} \cdot t) \cdot \text{L2V}(\rho(0))$ or equivalently $\rho(t) = \text{V2L}(\exp(\mathbf{M} \cdot t) \cdot \text{L2V}(\rho(0)))$ in a closed form. It can be obtained in polynomial time by the standard method [27].

2.3 Number Theory

► **Definition 1.** A number α is algebraic, denoted by $\alpha \in \mathbb{A}$, if there is a nonzero \mathbb{Q} -polynomial $f_\alpha(z)$ of least degree, satisfying $f_\alpha(\alpha) = 0$; otherwise α is transcendental.

In the above definition, such a polynomial $f_\alpha(z)$ is called the *minimal polynomial* of α . The degree D of α is $\deg_z(f_\alpha)$. The standard encoding of α is the minimal polynomial f_α plus an isolation disk in the complex plane that distinguishes α from other roots of f_α .

► **Definition 2.** Let μ_1, \dots, μ_m be irrational complex numbers. Then the field extension $\mathbb{Q}(\mu_1, \dots, \mu_m) : \mathbb{Q}$ is the smallest set that contains μ_1, \dots, μ_m and is closed under arithmetic operations, i.e. addition, subtraction, multiplication and division.

Here those irrational complex numbers μ_1, \dots, μ_m are called the generators of the field extension. A field extension is *simple* if it has only one generator. For instance, the field extension $\mathbb{Q}(\sqrt{2}) : \mathbb{Q}$ is exactly the set $\{a + b\sqrt{2} : a, b \in \mathbb{Q}\}$.

► **Lemma 3** ([32, Algorithm 2]). *Let α_1 and α_2 be two algebraic numbers of degrees D_1 and D_2 , respectively. There is an algebraic number μ of degree at most D_1D_2 , such that the field extension $\mathbb{Q}(\mu) : \mathbb{Q}$ is exactly $\mathbb{Q}(\alpha_1, \alpha_2) : \mathbb{Q}$.*

For a collection of algebraic numbers $\alpha_1, \dots, \alpha_m$ appearing in the input instance, by repeatedly applying this lemma, we can obtain a simple field extension $\mathbb{Q}(\mu) : \mathbb{Q}$ that can span all $\alpha_1, \dots, \alpha_m$.

► **Lemma 4** ([11, Corollary 4.1.5]). *Let α be an algebraic number of degree D , and $g(z)$ an \mathbb{A} -polynomial with degree D_g and coefficients taken from $\mathbb{Q}(\alpha) : \mathbb{Q}$. There is a \mathbb{Q} -polynomial $f(z)$ of degree at most DD_g , such that roots of $g(z)$ are those of $f(z)$.*

The above lemma entails that roots of any \mathbb{A} -polynomial are also algebraic.

► **Theorem 5** (Lindemann [7, Theorem 1.4]). *For any nonzero algebraic numbers β_1, \dots, β_m and any distinct algebraic numbers $\lambda_1, \dots, \lambda_m$, the sum $\sum_{i=1}^m \beta_i e^{\lambda_i}$ with $m \geq 1$ is nonzero.*

► **Conjecture 6** (Schanuel [2]). *Let $\lambda_1, \dots, \lambda_m$ be \mathbb{Q} -linearly independent complex numbers. Then the field extension $\mathbb{Q}(\lambda_1, e^{\lambda_1}, \dots, \lambda_m, e^{\lambda_m}) : \mathbb{Q}$ has transcendence degree at least m .*

Let $\mathbb{A}[z_1, \dots, z_m]$ denote the ring that contains all \mathbb{A} -polynomials in variables z_1, \dots, z_m . Assuming Schanuel's conjecture, we could get:

► **Corollary 7** ([9, Proposition 5]). *Let $\lambda_1, \dots, \lambda_m$ be \mathbb{Q} -linearly independent algebraic numbers. Then two co-prime elements φ_1 and φ_2 in the ring $\mathbb{A}[t, \exp(\lambda_1 t), \dots, \exp(\lambda_m t)]$ have no common root except for 0.*

3 Quantum Continuous-Time Markov Chain

In this section, we propose the model of quantum continuous-time Markov chain (QCTMC). We will reveal that it extends the classical continuous-time Markov chain (CTMC). To show the practical usefulness, an example is further provided for modelling open quantum walk.

For the sake of clarity, we start with the QCTMC without classical states:

- **Definition 8.** *A quantum continuous-time Markov chain \mathfrak{Q} is a pair $(\mathcal{H}, \mathcal{L})$, in which*
- \mathcal{H} is the Hilbert space,
 - \mathcal{L} is the transition generator function given by a Hermitian operator \mathbf{H} and a finite set of linear operators \mathbf{L}_j on \mathcal{H} .

Usually, a density operator $\rho(0) \in \mathcal{D}$ is appointed as the initial state of \mathfrak{Q} .

In the model, the transition generator function \mathcal{L} gives rise to a *universal* way to describe the behavior of the QCTMC, following the generality of the Lindblad's master equation. Thus the state $\rho(t)$ is given by the closed-form solution to Eq. (5), i.e., $V2L(\exp(\mathbf{M} \cdot t) \cdot L2V(\rho(0)))$, where \mathbf{M} is the governing matrix for \mathcal{L} , is a computable function from $\mathbb{R}_{\geq 0}$ to $\mathbb{C}^{d \times d}$. We notice that $0 \leq \text{tr}(\mathbf{P}\rho(t)) \leq 1$ holds for any projector \mathbf{P} on \mathcal{H} , as $\rho(t)$ is a density operator on \mathcal{D} . Considering computability, the entries of \mathbf{H} , \mathbf{L}_j and $\rho(0)$ are supposed to be algebraic.

Next, we equip the QCTMC in Definition 8 with finitely many classical states.

- **Definition 9.** *A quantum continuous-time Markov chain \mathfrak{Q} with a finite set S of classical states is a pair $(\mathcal{H}_{\text{cq}}, \mathcal{L})$, in which*

- $\mathcal{H}_{\text{cq}} := \mathcal{C} \otimes \mathcal{H}$ is the classical-quantum system with $\mathcal{C} = \text{span}(\{|s\rangle : s \in S\})$, and
- \mathcal{L} is the transition generator function given by a Hermitian operator \mathbf{H} and a finite set of linear operators \mathbf{L}_j on the enlarged \mathcal{H}_{cq} .

Usually, a density operator $\rho(0) \in \mathcal{D}_{\mathcal{H}_{\text{cq}}}$ is appointed as the initial state of \mathfrak{Q} .

In fact, the models in Definitions 8 and 9 have the same expressibility: The QCTMC in Definition 8 can be obtained by setting the singleton state set $S = \{s\}$ of the QCTMC in Definition 9; conversely, the QCTMC in Definition 9 can be obtained by setting the Hilbert space as \mathcal{H}_{c_q} of the QCTMC in Definition 8. Hence, we can freely choose one of the two definitions for convenience. As an immediate result, using Definition 9, we can easily see that the QCTMC extends the CTMC by the following lemma:

► **Lemma 10.** *Given a CTMC $\mathfrak{C} = (S, \mathbf{Q})$, it can be modelled by a QCTMC $\mathfrak{Q} = (\mathcal{C} \otimes \mathcal{H}, \mathcal{L})$ with $\mathcal{C} = \text{span}(\{|s\rangle : s \in S\})$ and $\dim(\mathcal{H}) = 1$.*

Proof. It suffices to show that the states of a CTMC \mathfrak{C} can be obtained by those of some QCTMC \mathfrak{Q} . The state $\mathbf{x} = (x_s)_{s \in S}$ of \mathfrak{C} is given by the dynamical system $\mathbf{x}'(t) = \mathbf{x}(t) \cdot \mathbf{Q}$ or equivalently its closed-form solution $\mathbf{x}(t) = \mathbf{x}(0) \cdot \exp(\mathbf{Q} \cdot t)$, where $\mathbf{x}(0)$ is a row vector interpreted as the initial state. We construct the QCTMC by setting Hermitian operator $\mathbf{H} = 0$ and linear operators $\mathbf{L}_{s,t} = |t\rangle\langle s| \otimes \mathbf{Q}[s, t]$ in \mathcal{Q} for each pair $s, t \in S$. It is not hard to validate that the state $\rho(t)$ of \mathfrak{Q} is $\text{diag}(\mathbf{x}(t))$ of \mathfrak{C} , thus the lemma follows. ◀

► **Example 11** (Open Quantum Walk [35, 36]). Open quantum walk (OQW) is a quantum analogy of random walk, whose system evolution interacts with environment. For the sake of clarity, we suppose that a particle walking along the 2-dimensional hypercube is shown in Figure 1. The position set is $S = \{s_{00}, s_{01}, s_{10}, s_{11}\}$, where s_{00} denotes the starting position, a.k.a. the entrance, s_{01} and s_{10} denote transient positions, and s_{11} denotes the exit, an absorbing boundary. The direction set is $\{F, S\}$, where F means the particle takes the external transition along the first coordinate, while S means the particle takes the external transition along the second coordinate. The particle will choose a direction at every moment by the inner quantum “coin-tossing” before being absorbed. This action is implemented by the Hadamard operator $H = |+\rangle\langle F| + |-\rangle\langle S|$ with $|\pm\rangle = (|F\rangle \pm |S\rangle)/\sqrt{2}$, which denotes a fair selection between F and S as the probability amplitudes of both directions are $\frac{1}{2} = (\pm 1/\sqrt{2})^2$.

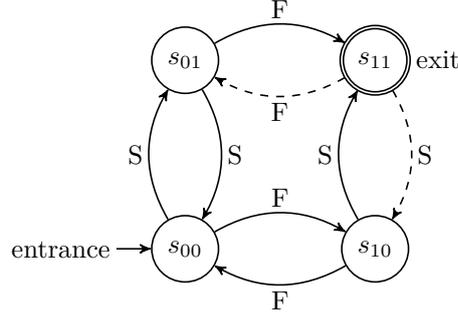
The OQW is modelled by a QCTMC $\mathfrak{Q}_1 = (\mathcal{C} \otimes \mathcal{H}, \mathcal{L})$ with $\mathcal{C} = \text{span}(\{|s\rangle : s \in S\})$, where each position in S represents a classical state, and the transition function \mathcal{L} is given by the Hermitian operator $\mathbf{H} = 0$ and the unique linear operator

$$\mathbf{L} = |s_{10}\rangle\langle s_{00}| \otimes |F\rangle\langle +| + |s_{01}\rangle\langle s_{00}| \otimes |S\rangle\langle -| + |s_{11}\rangle\langle s_{01}| \otimes |F\rangle\langle +| + |s_{00}\rangle\langle s_{01}| \otimes |S\rangle\langle -| + |s_{00}\rangle\langle s_{10}| \otimes |F\rangle\langle +| + |s_{11}\rangle\langle s_{10}| \otimes |S\rangle\langle -|.$$

For instance, when the particle is in the position $|s_{00}\rangle$, we first apply the quantum coin-tossing H to the state in \mathcal{H} , then we get the result F or S leading to the position $|s_{10}\rangle$ or $|s_{01}\rangle$. The composite operations are $(|F\rangle\langle F|)H = |F\rangle\langle +|$ and $(|S\rangle\langle S|)H = |S\rangle\langle -|$, which makes up the first two terms in the above \mathbf{L} .

From \mathbf{L} , we could get the governing matrix $\mathbf{M} = \mathbf{L} \otimes \mathbf{L}^* - \frac{1}{2}\mathbf{L}^\dagger \mathbf{L} \otimes \mathbf{I} - \frac{1}{2}\mathbf{I} \otimes \mathbf{L}^\dagger \mathbf{L}$. Let $\rho(0) = |s_{00}\rangle\langle s_{00}| \otimes |F\rangle\langle F|$ be an initial state. Then the states $\rho(t)$ of the OQW could be computed as $\text{V2L}(\exp(\mathbf{M} \cdot t) \cdot \text{L2V}(\rho(0)))$ (we omit the detailed value due to space limit). ◻

As an absorbing boundary s_{11} is introduced here, two transitions from it (in dashed line) do not exist anymore. The absorbing boundary makes it impossible to characterise the system evolution by a closed system, i.e., using some Hermitian operator \mathbf{H} only, as usual in [36]. Fortunately, we could characterise it by an open system, i.e., using the linear operators \mathbf{L}_j .



■ **Figure 1** A sample OQW with an absorbing boundary.

4 Signal Temporal Logic

Here we recall the signal temporal logic (STL) [33], which is widely used to express real-time properties. Using it we could specify richer properties of QCTMC than linear temporal logic (LTL) and computation tree logic (CTL) in the time-bounded fragment.

► **Definition 12.** *The syntax of the STL formulas are defined as follows:*

$$\phi := \Phi \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \text{U}^{\mathcal{I}} \phi_2$$

in which the atomic propositions Φ , interpreted as signals, are of the form $p(\mathbf{x}) \in \mathbb{I}$ where p is a \mathbb{Q} -polynomial in $\mathbf{x} = (x_s)_{s \in S}$ and \mathbb{I} is a rational interval, and \mathcal{I} is a finite time interval. Here U is called the until operator, and $\phi_1 \text{U}^{\mathcal{I}} \phi_2$ is the until formula.

► **Definition 13.** *The semantics of the STL formulas interpreted on a QCTMC Ω in Definition 9 are given by the satisfaction relation \models :*

$$\begin{aligned} \rho(t) \models \Phi & && \text{if } p(\mathbf{x}) \in \mathbb{I} \text{ holds with } x_s = \text{tr}(\mathbf{P}_s(\rho(t))), \\ \rho(t) \models \neg\phi & && \text{if } \rho(t) \not\models \phi, \\ \rho(t) \models \phi_1 \wedge \phi_2 & && \text{if } \rho(t) \models \phi_1 \wedge \rho(t) \models \phi_2, \\ \rho(t) \models \phi_1 \text{U}^{\mathcal{I}} \phi_2 & && \text{if there exists a real number } t' \in \mathcal{I}, \\ & && \text{such that } \forall t_1 \in [t, t+t') : \rho(t_1) \models \phi_1 \text{ and } \rho(t+t') \models \phi_2, \end{aligned}$$

where \mathbf{P}_s is the projector $|s\rangle\langle s| \otimes \mathbf{I}$ onto classical state s .

From the semantics, we can see that $\Phi_1 \text{U}^{\mathcal{I}_1} (\Phi_2 \text{U}^{\mathcal{I}_2} \Phi_3)$ and $(\Phi_1 \text{U}^{\mathcal{I}_1} \Phi_2) \text{U}^{\mathcal{I}_2} \Phi_3$ have different meanings: The former formula requires there exist $t' \in \mathcal{I}_1$ and $t'' \in \mathcal{I}_2$ such that $\forall t_1 \in [t, t+t') : \rho(t_1) \models \Phi_1$, $\forall t_2 \in [t+t', t+t'+t'') : \rho(t_2) \models \Phi_2$ and $\rho(t+t'+t'') \models \Phi_3$; while the latter formula requires there exists a $t'' \in \mathcal{I}_2$ such that $\rho(t+t'') \models \Phi_3$ and for each $t_1 \in [t, t+t'')$, there exists a $t' \in \mathcal{I}_1$ such that $\forall t_2 \in [t_1, t_1+t')$: $\rho(t_2) \models \Phi_1$ and $\rho(t_1+t') \models \Phi_2$. We usually use the *parse tree* to clarify the structure of an STL formula ϕ .

The logic is very generic. STL has more expressive atomic propositions than LTL, as **true** $\equiv p(\mathbf{x}) \in (-\infty, +\infty)$ and **false** $\equiv p(\mathbf{x}) \in \emptyset$. CTL has a two-stage syntax consisting of state and path formulas. Negation and conjunction are allowed in only state formulas, not path ones. Whereas, STL allows negation and conjunction in any subformulas. Besides the standard Boolean calculus, we can easily obtain a few derivations: $\diamond^{\mathcal{I}} \phi \equiv \text{true} \text{U}^{\mathcal{I}} \phi$, $\square^{\mathcal{I}} \phi \equiv \neg(\diamond^{\mathcal{I}} \neg\phi)$, and $\phi_1 \text{R}^{\mathcal{I}} \phi_2 \equiv \neg(\neg\phi_1 \text{U}^{\mathcal{I}} \neg\phi_2)$ where R is the release operator.

► **Example 14.** Consider the open quantum walk Ω_1 described in Example 11. It is not hard to get the probabilities of the particle staying respectively in $s_{00}, s_{01}, s_{10}, s_{11}$ as follows:

$$\begin{aligned} x_{0,0} &= \text{tr}(\mathbf{P}_{s_{0,0}}(\rho(t))) = \frac{1}{2} \exp(-\frac{2+\sqrt{2}}{2}t) + \frac{1}{2} \exp(-\frac{2-\sqrt{2}}{2}t), \\ x_{0,1} &= \text{tr}(\mathbf{P}_{s_{0,1}}(\rho(t))) = -\frac{\sqrt{2}}{4} \exp(-\frac{2+\sqrt{2}}{2}t) + \frac{\sqrt{2}}{4} \exp(-\frac{2-\sqrt{2}}{2}t) \\ &\quad + \frac{1}{4} [\exp(-\frac{3}{2}t) - \exp(-\frac{1}{2}t)] \cos(\frac{1}{2}t) + \frac{1}{4} [\exp(-\frac{3}{2}t) + \exp(-\frac{1}{2}t)] \sin(\frac{1}{2}t), \\ x_{1,0} &= \text{tr}(\mathbf{P}_{s_{1,0}}(\rho(t))) = -\frac{\sqrt{2}}{4} \exp(-\frac{2+\sqrt{2}}{2}t) + \frac{\sqrt{2}}{4} \exp(-\frac{2-\sqrt{2}}{2}t) \\ &\quad - \frac{1}{4} [\exp(-\frac{3}{2}t) - \exp(-\frac{1}{2}t)] \cos(\frac{1}{2}t) - \frac{1}{4} [\exp(-\frac{3}{2}t) + \exp(-\frac{1}{2}t)] \sin(\frac{1}{2}t), \\ x_{1,1} &= \text{tr}(\mathbf{P}_{s_{1,1}}(\rho(t))) = 1 + \frac{-1+\sqrt{2}}{2} \exp(-\frac{2+\sqrt{2}}{2}t) - \frac{1+\sqrt{2}}{2} \exp(-\frac{2-\sqrt{2}}{2}t). \end{aligned}$$

We can see that the probability of exiting via s_{11} would be one as t approaches infinity. A further question is asking how fast the particle would reach the exit. This is actually a question about convergence performance. It is worth monitoring the critical moment at which the probability of exiting via s_{11} equals that of staying at transient positions s_{01}, s_{10} . A requirement to be studied is like the following property.

Property A: At each moment $t \in [0, 5]$, whenever the probability of staying at s_{01} or s_{10} is greater than $\frac{1}{5}$, the probability of exiting via s_{11} would exceed that of staying at s_{01} or s_{10} within the coming one unit of time.

We formally specify it with the rich STL formula

$$\phi_1 \equiv \square^{\mathcal{I}_1} (\Phi_1 \rightarrow \diamond^{\mathcal{I}_2} \Phi_2) \equiv \neg(\text{true} \cup^{\mathcal{I}_1} (\Phi_1 \wedge \neg(\text{true} \cup^{\mathcal{I}_2} \Phi_2))),$$

where $\mathcal{I}_1 = [0, 5], \mathcal{I}_2 = [0, 1]$ are time intervals and $\Phi_1 \equiv x_{0,1} + x_{1,0} > \frac{1}{5}, \Phi_2 \equiv x_{1,1} \geq x_{0,1} + x_{1,0}$ are atomic propositions, a.k.a. signals. \lrcorner

5 Solving Atomic Propositions

As a basic step to decide the STL formula, we need to solve the atomic proposition Φ . That is, we will compute all solutions w.r.t. t , in which $\rho(t) \models \Phi$ holds. We achieve it by a reduction to the real root isolation for a class of real-valued functions, *exponential polynomials*. Although real roots of exponential polynomials have been studied in many existing literature [1, 9, 26], the ones to be isolated in this paper involve the complicated complex exponents. So we develop a state-of-the-art real root isolation for them, whose completeness is established on Conjecture 6.

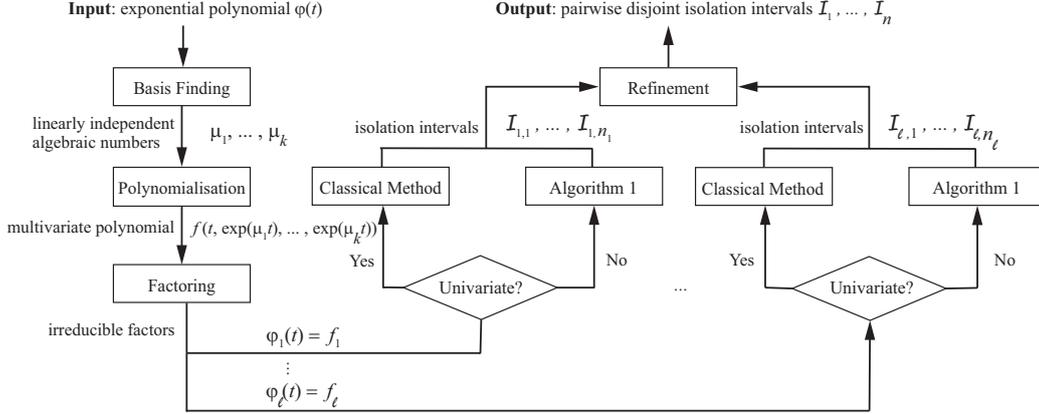
Given an atomic proposition $\Phi \equiv p(\mathbf{x}) \in \mathbb{I}$ (assuming that \mathbb{I} is bounded), we would like to determine the algebraic structure of

$$\varphi(t) = (p(\mathbf{x}(t)) - \inf \mathbb{I})(p(\mathbf{x}(t)) - \sup \mathbb{I}), \quad (6)$$

with which we will design an algorithm for solving $\Phi \equiv p(\mathbf{x}) \in \mathbb{I}$. The structure of $\varphi(t)$ depends on that of $x_s(t) = \text{tr}(\mathbf{P}_s(\rho(t)))$. We claim that each entry of $\rho(t)$ is of the exponential polynomial form

$$\beta_1(t) \exp(\alpha_1 t) + \beta_2(t) \exp(\alpha_2 t) + \cdots + \beta_m(t) \exp(\alpha_m t), \quad (7)$$

where $\beta_1(t), \dots, \beta_m(t)$ are nonzero \mathbb{A} -polynomials and $\alpha_1, \dots, \alpha_m$ are distinct algebraic numbers. It follows the facts:



■ **Figure 2** Flow chart of the whole isolation procedure.

1. The governing matrix \mathbf{M} in the state $\rho(t) = \mathbf{V}2\mathbf{L}(\exp(\mathbf{M} \cdot t) \cdot \mathbf{L}2\mathbf{V}(\rho(0)))$ of the QCTMC takes algebraic numbers as entries.
2. The characteristic polynomial of \mathbf{M} is an \mathbb{A} -polynomial. The eigenvalues $\alpha_1, \dots, \alpha_m$ of \mathbf{M} are algebraic, as they are roots of that \mathbb{A} -polynomial by Lemma 4. Those eigenvalues make up all exponents in (7).
3. The entries of the matrix exponential $\exp(\mathbf{M} \cdot t)$ are in the form (7).

The same structure holds for $x_s(t)$, as $x_s(t) = \text{tr}(\mathbf{P}_s(\rho(t)))$ is simply a sum of some entries of $\rho(t)$. Furthermore, $\varphi(t)$ is also of the exponential polynomial form (7), since it is a \mathbb{Q} -polynomial in $\mathbf{x}(t) = (x_s(t))_{s \in S}$. If \mathbb{I} is unbounded from below (resp. above), the left (resp. right) factor could be removed from (6) for further consideration.

Next, we will isolate all real roots $\lambda_1, \dots, \lambda_n$ of $\varphi(t)$ in a bounded interval \mathcal{B} (to be specified in the next section). Before stating the core isolation algorithm – Algorithm 1, an overview of the isolation procedure is provided in Fig. 2. The instances to be treated can be roughly divided into two classes: one is trivial that can be solved by the classical methods, e.g., [12], for ordinary polynomials; the other is nontrivial that can be solved by Algorithm 1 but should meet three requirements of the input in Algorithm 1. After the preprocesses **Basis Finding**, **Polynomialisation** and **Factoring**, the two classes of instances can be separated and solved by the corresponding methods. In the end, the **Refinement** process outputs the pairwise disjoint isolation intervals.

The technical details along the flow chart in Fig. 2 are described below.

- **Basis Finding.** For a given set of algebraic numbers $\{\alpha_1, \dots, \alpha_m\}$ extracted from the exponents of the input exponential polynomial $\varphi(t)$, we can compute a simple extension $\mathbb{Q}(\mu) : \mathbb{Q} = \mathbb{Q}(\alpha_1, \dots, \alpha_m) : \mathbb{Q}$ by Lemma 3, such that each $\alpha_i = q_i(\mu)$ with $q_i \in \mathbb{Z}[x]$ and $\deg(q_i) < \deg(\mu)$; and further construct a \mathbb{Q} -linearly independent basis $\{\mu_1, \dots, \mu_k\}$ of those exponents $\{\alpha_1, \dots, \alpha_m\}$ by [26, Section 3], such that each α_i can be \mathbb{Z}^+ -linearly expressed by $\{\mu_1, \dots, \mu_k\}$.
- **Polynomialisation.** Thus we can get a polynomial representation $f(t, \exp(\mu_1 t), \dots, \exp(\mu_k t))$ of $\varphi(t)$, where f is a $(k+1)$ -variate polynomial with algebraic coefficients. That is, $\varphi(t)$ is obtained by substituting $t, \exp(\mu_1 t), \dots, \exp(\mu_k t)$ (as $k+1$ variables) into f .
- **Factoring.** Factoring $\varphi(t)$ into irreducible factors $\varphi_i(t)$ ($1 \leq i \leq \ell$) corresponds to factoring the $(k+1)$ -variate \mathbb{A} -polynomial f into irreducible factors f_i ($1 \leq i \leq \ell$), which has been implemented in polynomial time, e.g. [28].

- **Univariate?** If the irreducible exponential polynomial $\varphi_i(t)$ corresponds to a univariate polynomial f_i , isolating the real roots of $\varphi_i(t)$ can be treated by classical methods [12]; otherwise we will resort to Algorithm 1, for which we can infer:
 1. The exponential polynomial $\varphi(t)$ in the form (7) is plainly an *analytic* function that is infinitely differentiable; and it is a real-valued function, or equivalently the imaginary part of $\varphi(t)$ is identically zero, since each variable $x_s(t)$ is exactly the real-valued function $\text{tr}(\mathbf{P}_s \rho(t))$. The same holds for any factor $\varphi_i(t)$ of $\varphi(t)$, which ensures the first requirement of the input in Algorithm 1.
 2. By Theorem 5, thanks to the irreducibility of $\varphi_i(t)$, we have $\varphi_i(\lambda) \neq 0$ holds for any $\lambda \in \mathbb{A} \setminus \{0\}$, which ensures the second requirement of the input in Algorithm 1.
 3. By Conjecture 6 and Corollary 7, each irreducible factor $\varphi_i(t)$ and its derivative $\varphi_i'(t)$ are co-prime, and thus have no common real root except for 0, which ensures the last requirement of the input in Algorithm 1.
- **Refinement.** After performing Algorithm 1 with each individual irreducible factor $\varphi_i(t)$ of $\varphi(t)$, we would obtain a list of disjoint isolation intervals $\mathcal{I}_{i,1}, \dots, \mathcal{I}_{i,n_i}$. The isolation intervals of different irreducible factors may be overlapping. However, by Corollary 7 again, we have that each pair of co-prime factors of $\varphi(t)$ has no common real root except for 0. So all these isolation intervals $\mathcal{I}_{i,1}, \dots, \mathcal{I}_{i,n_i}$ ($1 \leq i \leq \ell$) can be further refined to be pairwise disjoint. That would be the complete list of isolation intervals $\mathcal{I}_1, \dots, \mathcal{I}_n$ for $\varphi(t)$, and thereby completes the whole isolation procedure.

■ **Algorithm 1** Real Root Isolation for a Real-valued Function.

$$\{\mathcal{I}_1, \dots, \mathcal{I}_n\} \Leftarrow \text{Isolate}(\varphi, \mathcal{I})$$

Input: $\varphi(t)$ is a real-valued function defined on a rational interval $\mathcal{I} = [l, u]$, satisfying:

1. $\varphi(t)$ is twice-differentiable,
2. $\varphi(t)$ has no rational root in \mathcal{I} , and
3. $\varphi(t)$ and $\varphi'(t)$ have no common real root in \mathcal{I} .

Output: $\mathcal{I}_1, \dots, \mathcal{I}_n$ are finitely many disjoint intervals, such that each contains exactly one real root of φ in \mathcal{I} , and together contain all.

- 1: compute an upper bound M of $\{|\varphi'(t)| : t \in \mathcal{I}\}$;
 - 2: compute an upper bound M' of $\{|\varphi''(t)| : t \in \mathcal{I}\}$;
 - 3: $i \leftarrow 0$, $N \leftarrow 2$ and $\delta \leftarrow (u - l)/N$; \triangleright Here $N > 1$ is a free parameter to indicate the number of subintervals to be split. We predefine it simply as 2.
 - 4: **while** $i \leq N$ **do**
 - 5: **if** $|\varphi(l + i\delta)| > M\delta$ **then** $i \leftarrow i + 1$; $\triangleright \varphi$ has no local real root
 - 6: **else if** $|\varphi(l + i\delta + \delta)| > M\delta$ **then** $i \leftarrow i + 2$; $\triangleright \varphi$ has no local real root
 - 7: **else if** $|\varphi'(l + i\delta)| \geq M'\delta$ **then** $\triangleright \varphi$ is locally monotonic
 - 8: **if** $\varphi(l + i\delta)\varphi(l + i\delta + \delta) < 0$ **then output** $(l + i\delta, l + i\delta + \delta)$;
 - 9: $i \leftarrow i + 1$;
 - 10: **else if** $|\varphi'(l + i\delta + \delta)| \geq M'\delta$ **then** $\triangleright \varphi$ is locally monotonic
 - 11: **if** $\varphi(l + i\delta)\varphi(l + i\delta + 2\delta) < 0$ **then output** $(l + i\delta, l + i\delta + 2\delta)$;
 - 12: $i \leftarrow i + 2$;
 - 13: **else**
 - 14: **Isolate** $(\varphi, [l + i\delta, l + i\delta + \delta])$;
 - 15: $i \leftarrow i + 1$.
-

Soundness. We justify three groups of treatment in the loop body in turn.

1. If the condition in Line 5 (resp. 6) holds, φ has no real root in the neighborhood centered at $l + i\delta$ (resp. $l + (i + 1)\delta$) with radius δ . So we exclude the neighborhood.
2. If the condition in Line 7 (resp. 10) holds, φ is monotonic in the neighborhood centered at $l + i\delta$ (resp. $l + (i + 1)\delta$) with radius δ . Then, if the condition in Line 8 (resp. 11) holds, i.e. φ has different signs at endpoints of the neighborhood, the unique real root in the neighborhood exists and should be output; otherwise we just exclude the neighborhood.
3. If it is not in the two decisive cases listed above, we perform Algorithm 1 recursively in a subinterval $[l + i\delta, l + (i + 1)\delta]$ of $[l, u]$. \lrcorner

Completeness. The termination of Algorithm 1 entails the completeness. In other words, it suffices to show that for any real-valued function φ that satisfies the requirements, Algorithm 1 can always output all real roots of φ within finitely many times of recursion. Since φ and φ' are real-valued functions defined on the closed and bounded interval \mathcal{I} and they have no common real root in \mathcal{I} , there is a positive constant τ , such that either $|\varphi| \geq \tau$ or $|\varphi'| \geq \tau$ holds everywhere of \mathcal{I} . Then, at any point c in \mathcal{I} , we can get a neighborhood with constant radius $rad := \min(\tau/M, \tau/M')$, in which either φ has no real root or is monotonic. So the two decisive cases must take place, provided that the subinterval has a length not greater than rad . It implies that the recursion depth of Algorithm 1 is bounded by $\lceil \log_2(\|\mathcal{I}\|/rad) \rceil$, where $\|\mathcal{I}\| = \sup \mathcal{I} - \inf \mathcal{I}$. Hence the termination is guaranteed. \lrcorner

After obtaining all real roots $\lambda_1, \dots, \lambda_n$ of $\varphi(t)$ in the bounded interval \mathcal{B} by Algorithm 1, we have that on each interval \mathcal{J}_i ($0 \leq i \leq n$) of the $n + 1$ intervals in $\mathcal{B} \setminus \{\lambda_1, \dots, \lambda_n\}$:

$$(\inf \mathcal{B}, \lambda_1), (\lambda_1, \lambda_2), \dots, (\lambda_{n-1}, \lambda_n), (\lambda_n, \sup \mathcal{B}), \quad (8)$$

$p(\mathbf{x}(t)) \in \mathbb{I}$ holds everywhere of $t \in \mathcal{J}_i$ or nowhere. Finally, we can obtain the desired solution set \mathbb{J} (to be used in the next section) of $p(\mathbf{x}(t)) \in \mathbb{I}$ by a finite union as follows

$$\bigcup_{\substack{0 \leq i \leq n \\ p(\mathbf{x}(\mathcal{J}_i)) \subseteq \mathbb{I}}} \mathcal{J}_i \cup \bigcup_{\substack{1 \leq i \leq n \\ p(\mathbf{x}(\lambda_i)) \in \mathbb{I}}} \{\lambda_i\}. \quad (9)$$

► **Example 15.** Reconsider Example 14. The exponential polynomial extracted from Φ_1 is

$$\varphi_1(t) = x_{0,1}(t) + x_{1,0}(t) - \frac{1}{5} = -\frac{1}{5} - \frac{\sqrt{2}}{2} \exp(-\frac{2+\sqrt{2}}{2}t) + \frac{\sqrt{2}}{2} \exp(-\frac{2-\sqrt{2}}{2}t),$$

and the exponential polynomial extracted from Φ_2 is

$$\varphi_2(t) = x_{1,1}(t) - x_{0,1}(t) - x_{1,0}(t) = 1 + (\sqrt{2} - \frac{1}{2}) \exp(-\frac{2+\sqrt{2}}{2}t) - (\sqrt{2} + \frac{1}{2}) \exp(-\frac{2-\sqrt{2}}{2}t).$$

To solve Φ_1 and Φ_2 in a bounded interval, say $\mathcal{B} = [0, 6]$, we need to determine the real roots of φ_1 and φ_2 . Since both φ_1 and φ_2 are irreducible and their corresponding polynomial representations are bivariate, they have no rational root, repeated real root, nor common real root. After invoking Algorithm 1 on φ_1 with \mathcal{B} , we obtain two isolation intervals $[0, \frac{25}{64}]$ (containing real root $\lambda_1 \approx 0.352097$) and $[\frac{275}{64}, \frac{575}{128}]$ (containing $\lambda_2 \approx 4.49181$), which could be easily refined up to any precision. For φ_2 , as $\varphi_2(0) = 0$, we make a slight shift on the left endpoint of \mathcal{B} under consideration, e.g., $[\frac{1}{1000}, 6]$. After invoking Algorithm 1 on φ_2 with $[\frac{1}{1000}, 6]$, we could get the unique isolation intervals $[\frac{125059}{64000}, \frac{275117}{128000}]$, which contains the real root $\lambda_3 \approx 2.14897$. The three isolation intervals are pairwise disjoint.

Finally, we have that the solution set of Φ_1 is (λ_1, λ_2) in \mathcal{B} , and the solution set of Φ_2 is $\{0\} \cup [\lambda_3, 6]$. \lrcorner

Under Conjecture 6, we get the following computability result:

► **Lemma 16.** *The atomic propositions in STL are solvable over QCTMCs.*

The solvability here means that given an atomic proposition Φ and a bounded interval \mathcal{B} , we can always compute all solutions \mathbb{J} of Φ in \mathcal{B} , which consists of a finite number of pairwise disjoint solution intervals \mathcal{J}_i with computable real numbers as endpoints. The solving procedure is an *exact* one, i.e., no numerical error is allowed. Although it is efficient in practice, its complexity is still an open problem, as is left in existing literature [9, 26]. To pursue the guarantee of polynomial time, we could allow some numerical errors, which results in an *approximate* solving procedure.

In fact, Conjecture 6 is a powerful tool to treat roots of the general exponential polynomial. For some special subclasses of exponential polynomials, there are solid theorems to treat them: one is Theorem 5 that has been employed [1] for the exponential polynomials in the ring $\mathbb{Q}[t, \exp(t)]$, the other is the Gelfond–Schneider theorem employed [26, Subsection 4.1] for the exponential polynomials in $\mathbb{Q}[\exp(\mu_1 t), \exp(\mu_2 t)]$ where μ_1 and μ_2 are *two* \mathbb{Q} -linear independent *real* algebraic numbers. In Example 15, the Gelfond–Schneider theorem is sufficient to treat roots of the exponential polynomials φ_1 and φ_2 , thus the termination is guaranteed. Algorithm 1 can isolate roots of elements in $\mathbb{Q}[t, \exp(\mu_1 t), \dots, \exp(\mu_k t)]$ for *arbitrarily many* \mathbb{Q} -linear independent *complex* algebraic numbers μ_1, \dots, μ_k . Additionally, Conjecture 6 has been employed to isolate simple roots of more expressible functions than exponential polynomials [38, 39], but fails to find repeated roots. Hence, this paper makes a trade-off between the expressibility of functions and the completeness of methodologies.

6 Checking STL Formulas

In the previous section, we have solved atomic propositions. Now we consider the general STL formula ϕ . For a given formula ϕ , we compute the so-called *post-monitoring period* $\text{mnt}(\phi)$, independent from the initial time t_0 , such that the truth of $\rho(t_0) \models \phi$ could be affected by those $\rho(t)$ with $t \in [t_0, t_0 + \text{mnt}(\phi)]$. Then we decide ϕ with a bottom-up fashion. The complexity turns out to be linear in the size $\|\phi\|$ of the input STL formula ϕ , which is defined as the number of logic connectives in ϕ as standard.

Given an STL formula ϕ , we need to post-monitor a time period to decide the truth of $\rho(t_0) \models \phi$ at an initial time t_0 , especially for the until formula $\phi_1 \text{U}^{\mathcal{I}} \phi_2$. For example, according to the semantics of STL, to decide $\rho(t_0) \models \phi_2$ with $\phi_2 \equiv \Phi_1 \text{U}^{\mathcal{I}_1} (\Phi_2 \text{U}^{\mathcal{I}_2} \Phi_3)$, we have to monitor the states $\rho(t)$ from time t_0 to $t_0 + \sup \mathcal{I}_1 + \sup \mathcal{I}_2$. It inspires us to calculate the post-monitoring period $\text{mnt}(\phi)$ as

$$\text{mnt}(\phi) = \begin{cases} 0 & \text{if } \phi = \Phi, \\ \text{mnt}(\phi_1) & \text{if } \phi = \neg\phi_1, \\ \max(\text{mnt}(\phi_1), \text{mnt}(\phi_2)) & \text{if } \phi = \phi_1 \wedge \phi_2, \\ \sup \mathcal{I} + \max(\text{mnt}(\phi_1), \text{mnt}(\phi_2)) & \text{if } \phi = \phi_1 \text{U}^{\mathcal{I}} \phi_2. \end{cases} \quad (10)$$

► **Lemma 17.** *Given an STL formula ϕ , the satisfaction $\rho(t_0) \models \phi$ is entirely determined by the states $\rho(t)$ with $t_0 \leq t \leq t_0 + \text{mnt}(\phi)$.*

Proof. We discuss it upon the syntactical structure of the STL formula ϕ .

For the atomic proposition Φ , $\rho(t_0) \models \Phi$ is plainly determined by $\rho(t_0)$.

For the negation $\neg\phi_1$, if $\rho(t_0) \models \phi_1$ is determined by $\rho(t)$ with $t_0 \leq t \leq t_0 + \text{mnt}(\phi_1)$, so is $\rho(t_0) \models \neg\phi_1$.

For the conjunction $\phi_1 \wedge \phi_2$, if $\rho(t_0) \models \phi_1$ (resp. $\rho(t_0) \models \phi_2$) is determined by $\rho(t)$ with $t_0 \leq t \leq t_0 + \text{mnt}(\phi_1)$ (resp. $t_0 \leq t \leq t_0 + \text{mnt}(\phi_2)$), $\rho(t_0) \models \phi_1 \wedge \phi_2$ is determined by the union of those states, i.e. $\rho(t)$ with $t_0 \leq t \leq t_0 + \max(\text{mnt}(\phi_1), \text{mnt}(\phi_2))$.

For the until formula $\phi_1 \text{U}^{\mathcal{I}} \phi_2$, if $\rho(t_0) \models \phi_1$ (resp. $\rho(t_0) \models \phi_2$) is determined by $\rho(t)$ with $t_0 \leq t \leq t_0 + \text{mnt}(\phi_1)$ (resp. $t_0 \leq t \leq t_0 + \text{mnt}(\phi_2)$), $\rho(t_0) \models \phi_1 \text{U}^{\mathcal{I}} \phi_2$ is determined by $\rho(t)$ with $t_0 \leq t \leq t_0 + \sup \mathcal{I} + \max(\text{mnt}(\phi_1), \text{mnt}(\phi_2))$, where $\sup \mathcal{I}$ is caused by the admissible transition at the latest time in the until formula, since then we have to determine $\rho(t) \models \phi_1$ from time t_0 to $t_0 + \sup \mathcal{I}$ and determine $\rho(t_0 + \sup \mathcal{I}) \models \phi_2$. ◀

To decide $\rho(0) \models \phi$, our method is based on the parse tree \mathcal{T} of ϕ as follows.

Basically, for each leaf of \mathcal{T} that represents an atomic proposition Φ , we compute the solution set \mathbb{J} (possibly a union of maximal solution intervals \mathcal{J}) of Φ within the monitoring interval $\mathcal{B} := [0, \text{mnt}(\phi)]$ by Algorithm 1.

Inductively, for each intermediate node of \mathcal{T} that represents the subformula ψ of ϕ , we tackle it into three classes.

- If $\psi = \neg \phi_1$, supposing that \mathbb{J}_1 is the solution set of ϕ_1 , the solution set \mathbb{J}' of ψ is $\mathcal{B} \setminus \mathbb{J}_1$.
- If $\psi = \phi_1 \wedge \phi_2$, supposing that \mathbb{J}_1 (resp. \mathbb{J}_2) is the solution set of ϕ_1 (resp. ϕ_2), the solution set \mathbb{J}' of ψ is $\mathbb{J}_1 \cap \mathbb{J}_2$.
- If $\psi = \phi_1 \text{U}^{\mathcal{I}} \phi_2$, supposing that \mathbb{J}_1 (resp. \mathbb{J}_2) is the solution set of ϕ_1 (resp. ϕ_2), the solution set \mathbb{J}' of ψ is

$$\{t : (t' \in \mathcal{I}) \wedge ([t, t+t'] \subseteq \mathbb{J}_1) \wedge (t+t' \in \mathbb{J}_2)\}.$$

directly from the semantics of the until formula $\phi_1 \text{U}^{\mathcal{I}} \phi_2$ in Definition 13, as

- $[t, t+t'] \subseteq \mathbb{J}_1$ if and only if $\forall t_1 \in [t, t+t'] : \rho(t_1) \models \phi_1$, and
- $t+t' \in \mathbb{J}_2$ if and only if $\rho(t+t') \models \phi_2$.

Note that the inductive steps of the above procedure do not generally produce all solutions of the subformula ψ in \mathcal{B} . Since by Lemma 17 $\rho(t_0) \models \psi$ is entirely determined by $\rho(t)$ with $t_0 \leq t \leq t_0 + \text{mnt}(\psi)$, the resulting solution set \mathbb{J}' contains all solutions of ψ in $[0, \text{mnt}(\phi) - \text{mnt}(\psi)]$ and possibly misses some solutions in the right subinterval $(\text{mnt}(\phi) - \text{mnt}(\psi), \text{mnt}(\phi)]$. Anyway, the subinterval $[0, \text{mnt}(\phi) - \text{mnt}(\psi)]$ has the left-closed endpoint 0, which suffices to decide $\rho(0) \models \phi$.

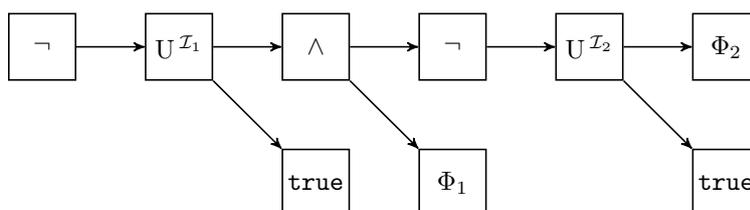
With a bottom-up fashion, we could eventually get the solution set \mathbb{J} of ϕ , by which $\rho(0) \models \phi$ can be decided to be true if and only if $0 \in \mathbb{J}$. Overall, the procedure costs $\|\phi\|$ times of the interval operations and at most $\|\phi\|$ times of calling Algorithm 1 for getting the solution set of an atomic proposition. That is, the query complexity of model checking STL formulas is linear in $\|\phi\|$ by calling Algorithm 1. The query complexity addresses the issue of the number of calls to a black box routine with unknown complexities and is commonly used in quantum computing, where the routine is usually called an oracle. For example, oracles can be a procedure of encoding an entry of a matrix into a quantum state [25] or a quantum circuit of preparing a specific quantum state [22].

► **Example 18.** For the STL formula $\phi_1 \equiv \neg(\text{true} \text{U}^{\mathcal{I}_1} (\Phi_1 \wedge \neg(\text{true} \text{U}^{\mathcal{I}_2} \Phi_2)))$, the post-monitoring period $\text{mnt}(\phi_1)$ is $\sup \mathcal{I}_1 + \sup \mathcal{I}_2 = 6$ by Eq. (10), implying $\mathcal{B} = [0, 6]$ as used in Example 15. We construct the parse tree of ϕ_1 in Figure 3. Based on it, we could calculate the solution sets of all nodes in a bottom-up fashion:

- Basically, the solution set for Φ_1 is (λ_1, λ_2) where $\lambda_1 \approx 0.352097$ and $\lambda_2 \approx 4.49181$, the solution set for Φ_2 is $\{0\} \cup [\lambda_3, 6]$ where $\lambda_3 \approx 2.14897$, and the solution set for **true** is plainly $[0, 6]$. The post-monitoring periods of the three distinct leaves are 0. Since \mathcal{B} is $[0, 6]$, we have got all the solutions in $[0, 6]$.

- The solution set for $\text{true} \mathcal{U}^{\mathcal{I}_2} \Phi_2$ is $[\lambda_3 - 1, 6]$, and that for the negation is $[0, \lambda_3 - 1)$. The post-monitoring periods of the two nodes are 1, thus we have got all solutions in $[0, 5]$.
- The solution set for $\Phi_1 \wedge \neg(\text{true} \mathcal{U}^{\mathcal{I}_2} \Phi_2)$ is $(\lambda_1, \lambda_3 - 1)$. Its post-monitoring period is 1, thus we have got all solutions in $[0, 5]$.
- Finally, the solution set for $\text{true} \mathcal{U}^{\mathcal{I}_1} (\Phi_1 \wedge \neg(\text{true} \mathcal{U}^{\mathcal{I}_2} \Phi_2))$ is $[0, \lambda_3 - 1)$, and that for the negation (the root, representing the whole STL formula ϕ_1) is $[\lambda_3 - 1, 6]$. Their post-monitoring periods are 6, thus we have got the solution in $[0, 0]$.

Since $0 \notin [\lambda_3 - 1, 6]$, we can decide $\rho(0) \models \phi_1$ to be false. Hence the particle walking along Figure 1 does not satisfy the desired convergence performance – Property A. \square



■ **Figure 3** Parse tree of the STL formula ϕ_1

Finally, under Conjecture 6, we obtain the main result:

► **Theorem 19.** *The STL formulas are decidable over QCTMCs.*

7 Concluding Remarks

In this paper, we introduced the model of QCTMC that extends CTMC, and established the decidability of the STL formulas over it. To this goal, we firstly solved the atomic propositions in STL by real root isolation of a wide class of exponential polynomials, whose completeness was based on Schanuel’s conjecture. Then we decided the general STL formula using interval operations with a bottom-up fashion, whose query complexity turned out to be linear in the size of the input formula by calling the developed state-of-the-art real root isolation routine. We demonstrated our method by a running example of an open quantum walk. For future work, we would like to explore the following aspects:

- how to apply the proposed method to verify non-Markov models in the real world [35];
- how to design an efficient numerical approximation of the exact method in this paper;
- and checking other formal logics, e.g. [3], over the QCTMC.

References

- 1 M. Achatz, S. McCallum, and V. Weispfenning. Deciding polynomial–exponential problems. In *Proc. 33rd International Symposium on Symbolic and Algebraic Computation, ISSAC 2008*, pages 215–222. ACM Press, 2008.
- 2 J. Ax. On Schanuel’s conjectures. *Annals of Mathematics*, 93(2):252–268, 1971.
- 3 A. Aziz, K. Sanwal, V. Singhal, and R. Brayton. Verifying continuous time Markov chains. In *Computer Aided Verification: 8th International Conference, CAV’96*, volume 1102 of *LNCS*, pages 269–276. Springer, 1996.
- 4 A. Aziz, K. Sanwal, V. Singhal, and R. Brayton. Model-checking continuous-time Markov chains. *ACM Transactions on Computational Logic*, 1(1):162–170, 2000.

- 5 C. Baier, B. Haverkort, H. Hermanns, and J.-P. Katoen. Model-checking algorithms for continuous-time Markov chains. *IEEE Transactions on Software Engineering*, 29(6):524–541, 2003.
- 6 C. Baier, J.-P. Katoen, and H. Hermanns. Approximate symbolic model checking of continuous-time Markov chains. In *CONCUR'99 Concurrency Theory*, volume 1664 of *LNCS*, pages 146–161. Springer, 1999.
- 7 A. Baker. *Transcendental Number Theory*. Cambridge University Press, 1975.
- 8 B. Baumgartner and H. Narnhofer. The structures of state space concerning quantum dynamical semigroups. *Reviews in Mathematical Physics*, 24(02):article no. 1250001, 2012.
- 9 V. Chonev, J. Ouaknine, and J. Worrell. On the Skolem problem for continuous linear dynamical systems. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016*, volume 55 of *LIPICs*, pages 100:1–100:13. Schloss Dagstuhl, 2016.
- 10 G. I. Cirillo and F. Ticozzi. Decompositions of Hilbert spaces, stability analysis and convergence probabilities for discrete-time quantum dynamical semigroups. *Journal of Physics A: Mathematical and Theoretical*, 48(8):article no. 085302, 2015.
- 11 H. Cohen. *A Course in Computational Algebraic Number Theory*. Springer, 1996.
- 12 G. E. Collins and R. Loos. Polynomial real root isolation by differentiation. In *Proc. SYMSAC 1976*, pages 15–25. ACM Press, 1976.
- 13 C. Dehnert, S. Junges, J.-P. Katoen, and M. Volk. A STORM is coming: A modern probabilistic model checker. In *Computer Aided Verification - 29th International Conference, CAV 2017, Part II*, volume 10427 of *LNCS*, pages 592–600. Springer, 2017.
- 14 Y. Feng, E. M. Hahn, A. Turrini, and S. Ying. Model checking ω -regular properties for quantum Markov chains. In *28th International Conference on Concurrency Theory, CONCUR 2017*, volume 85 of *LIPICs*, pages 35:1–35:16. Schloss Dagstuhl, 2017.
- 15 Y. Feng, N. Yu, and M. Ying. Model checking quantum Markov chains. *Journal of Computer and System Sciences*, 79(7):1181–1198, 2013.
- 16 S. J. Gay, R. Nagarajan, and N. Papanikolaou. Probabilistic model-checking of quantum protocols. In *Proc. 2nd International Workshop on Developments in Computational Models*, 2006. available at [arXiv:quant-ph/0504007](https://arxiv.org/abs/quant-ph/0504007).
- 17 S. J. Gay, R. Nagarajan, and N. Papanikolaou. QMC: A model checker for quantum systems. In *Computer Aided Verification, 20th International Conference, CAV 2008*, volume 5123 of *LNCS*, pages 543–547. Springer, 2008.
- 18 S. Goldstein, J. L. Lebowitz, R. Tumulka, and N. Zanghì. Long-time behavior of macroscopic quantum systems. *The European Physical Journal H*, 35(2):173–200, 2010.
- 19 V. Gorini, A. Kossakowski, and E. C. G. Sudarshan. Completely positive dynamical semigroups of n -level systems. *Journal of Mathematical Physics*, 17(5):821–825, 1976.
- 20 J. Guan, Y. Feng, A. Turrini, and M. Ying. Model checking applied to quantum physics. *CoRR*, abs/1902.03218, 2019. [arXiv:1902.03218](https://arxiv.org/abs/1902.03218).
- 21 J. Guan, Y. Feng, and M. Ying. Decomposition of quantum Markov chains and its applications. *Journal of Computer and System Sciences*, 95:55–68, 2018.
- 22 J. Guan, Q. Wang, and M. Ying. An HHL-based algorithm for computing hitting probabilities of quantum walks. *Quantum Information and Computation*, 2021(5&6):0395–0408, 2021.
- 23 J. Guan and N. Yu. Verification of continuous-time Markov chains. *CoRR*, abs/2004.08059, 2020. [arXiv:2004.08059](https://arxiv.org/abs/2004.08059).
- 24 E. M. Hahn, Y. Li, S. Schewe, A. Turrini, and L. Zhang. ISCASMC: A web-based probabilistic model checker. In *FM 2014: Formal Methods—19th International Symposium*, volume 8442 of *LNCS*, pages 312–317. Springer, 2014.
- 25 A. W. Harrow, A. Hassidim, and S. Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15):article no. 150502, 2009.
- 26 C.-C. Huang, J.-C. Li, M. Xu, and Z.-B. Li. Positive root isolation for poly-powers by exclusion and differentiation. *Journal of Symbol Computation*, 85:148–169, 2018.
- 27 T. Kailath. *Linear Systems*. Prentice Hall, 1980.

- 28 E. Kaltofen. Polynomial-time reductions from multivariate to bi- and univariate integral polynomial factorization. *SIAM Journal on Computing*, 14(2):469–489, 1985.
- 29 M. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of probabilistic real-time systems. In *Computer Aided Verification: 23rd International Conference, CAV 2011*, volume 6806 of *LNCS*, pages 585–591. Springer, 2011.
- 30 L. Li and Y. Feng. Quantum Markov chains: Description of hybrid systems, decidability of equivalence, and model checking linear-time properties. *Information and Computation*, 244:229–244, 2015.
- 31 G. Lindblad. On the generators of quantum dynamical semigroups. *Communications in Mathematical Physics*, 48(2):119–130, 1976.
- 32 R. Loos. Computing in algebraic extensions. In *Computer Algebra: Symbolic and Algebraic Computation*, pages 173–187. Springer, 1983.
- 33 O. Maler and D. Nickovic. Monitoring temporal properties of continuous signals. In *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, volume 3253 of *LNCS*, pages 152–166. Springer, 2004.
- 34 M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.
- 35 C. Pellegrini. Continuous time open quantum random walks and non-Markovian Lindblad master equations. *Journal of Statistical Physics*, 154:838–865, 2014.
- 36 I. Sinayskiy and F. Petruccione. Microscopic derivation of open quantum walks. *Physical Review A*, 92(3):article no. 032105, 2015.
- 37 W. J. Stewart. *Introduction to the Numerical Solution of Markov Chains*. Princeton University Press, 1994.
- 38 A. Strzeboński. Real root isolation for exp–log functions. In *Proc. 33rd. International Symposium on Symbolic and Algebraic Computation, ISSAC 2008*, pages 303–313. ACM Press, 2008.
- 39 A. Strzeboński. Real root isolation for tame elementary functions. In *Proc. 34th International Symposium on Symbolic and Algebraic Computation, ISSAC 2009*, pages 341–350. ACM Press, 2009.
- 40 M. M. Wolf. Quantum channels & operations: Guided tour. *Lecture notes available at <http://www-m5.ma.tum.de/foswiki/pubM>*, 5, 2012.
- 41 M. Xu, C.-C. Huang, and Y. Feng. Measuring the constrained reachability in quantum Markov chains. *Acta Informatica, online first*, 2021.
- 42 M. Xu, L. Zhang, D. N. Jansen, H. Zhu, and Z. Yang. Multiphase until formulas over Markov reward models: An algebraic approach. *Theoretical Computer Science*, 611:116–135, 2016.
- 43 M. Ying and Y. Feng. Model-checking quantum systems. *National Science Review*, 6(1):28–31, 2019.
- 44 M. Ying and Y. Feng. *Model Checking Quantum Systems: Principles and Algorithms*. Cambridge University Press, 2021.
- 45 S. Ying, Y. Feng, N. Yu, and M. Ying. Reachability probabilities of quantum Markov chains. In *CONCUR 2013: Concurrency Theory – 24th International Conference*, volume 8052 of *LNCS*, pages 334–348. Springer, 2013.
- 46 N. Yu. Quantum temporal logic. *CoRR*, abs/1908.00158, 2019. [arXiv:1908.00158](https://arxiv.org/abs/1908.00158).
- 47 N. Yu and M. Ying. Reachability and termination analysis of concurrent quantum programs. In M. Koutny and I. Ulidowski, editors, *CONCUR 2012 - Concurrency Theory - 23rd International Conference*, volume 7454 of *LNCS*, pages 69–83. Springer, 2012.
- 48 L. Zhang, D. N. Jansen, F. Nielson, and H. Hermanns. Automata-based CSL model checking. In *Automata, Languages and Programming: 38th International Colloquium, ICALP 2011, Part II*, volume 6756 of *LNCS*, pages 271–282. Springer, 2011.
- 49 L. Zhang, D. N. Jansen, F. Nielson, and H. Hermanns. Efficient CSL model checking using stratification. *Logical Methods in Computer Science*, 8(2:17):1–18, 2012.