arXiv:2307.03858v3 [quant-ph] 9 Jul 2024

Simulation-assisted Learning of Open Quantum Systems

Ke Wang and Xiantao Li

Department of Mathematics, Pennsylvania State University

Models for open quantum systems, which play important roles in electron transport problems and quantum computing, must take into account the interaction of the quantum system with the surrounding environment. Although such models can be derived in some special cases, in most practical situations, the exact models are unknown and have to be calibrated. This paper presents a learning method to infer parameters in Markovian open quantum systems from measurement data. One important ingredient in the method is a direct simulation technique of the quantum master equation, which is designed to preserve the completely-positive property with guaranteed accuracy. The method is particularly helpful in the situation where the time intervals between measurements are large. The approach is validated with error estimates and numerical experiments.

1 Introduction

Quantum learning has recently emerged as a field at the intersection of quantum physics and computer science [1, 2, 3]. Its primary focus is on the estimation and characterization of energy levels and interaction coefficients within a quantum system. Such effort has become increasingly crucial for characterizing, optimizing, and controlling quantum systems. One particularly explored area is Hamiltonian learning, where the emphasis is to infer the Hamiltonians [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17] , which ultimately determine the system's energy spectrum and dynamic behavior. Compared to direct simulation approaches, the learning process attempts to map observation data back to the model parameters. An important extension of Hamiltonian learning is the parameter identification of open quantum systems [18, 19, 20] , ones that continuously interact with their environment [21]. The ultimate goal is to reconstruct the interactions in the Lindblad-Gorini-Kossakowski-Sudarshan quantum master equation (QME) [22, 23]:

$$\frac{d}{dt}\rho = -i[H,\rho] + \sum_{j=1}^{N_V}(V_j\rho V_j^\dagger - \frac{1}{2}V_j^\dagger V_j\rho - \frac{1}{2}\rho V_j^\dagger V_j), \tag{1}$$

where H is the system Hamiltonian and V_j's are jump operators that arise from the interactions with the environment [21].

In practice, the learning tasks are usually formulated as an optimization problem, which is then solved by an iterative method, and as such, one needs to frequently access

Ke Wang: wangke.math@psu.edu
Xiantao Li: Xiantao.Li@psu.edu

the objective function, e.g., through the expectations of some observables. The ability to acquire such quantities might be limited by the measurement capabilities. Furthermore, the availability of the gradient of the objective function, a necessary ingredient for a fast optimization algorithm, might not be attainable through experiments either. In this paper, we propose a simulation-assisted method to address these issues. In contrast to the equation-based methods [18, 19] to identify Lindbladians in (1), we formulate a *trajectory-based* framework, where the observations $A(t) = \mathrm{tr}\left(Ae^{t\mathcal{L}}\rho(0)\right)$ are fit by simulating the QME (1). The main departure from equation-based methods [19, 24, 20] is that we do not assume access to the quantum system. We designed a classical learning algorithm with a given time series measurements as input. Namely, we do not assume the flexibility of choosing the time steps or the utilization of classical shadow tomography to make further measurements. For example, the efficiency in the sample complexity in the approach [19] requires choosing the initial and end time carefully. Part of the challenges comes from the presence of noise in the data, which complicates the numerical differentiation procedure. In our approach, the accuracy is controlled by chopping measurement intervals Δt into smaller steps δt, and in between we simulate the Lindblad dynamics with the proposed parameters.

Another notable difference from the equation-based approaches [18, 19] is that our optimization problem does not require the expectations associated with the right-hand side of Eq. (1), i.e., $\mathrm{tr}\left(OV_j\rho V_j^\dagger\right), \mathrm{tr}\left(OV_j^\dagger V_j\rho\right), \mathrm{tr}\left(O\rho V_j^\dagger V_j\right)$, and thus fewer features are needed from the data set.

To enable such a simulation-assisted approach, we propose a simulation algorithm, denoted here by $\mathcal{M}_{\delta t}(t)$, that has global error δt^2, in that $e^{\mathcal{L}t} - \mathcal{M}_{\delta t}(t) = \mathcal{O}\left(\delta t^2\right)$, we choose a semi-implicit algorithm, which is stable even when the coherent term in the QME (1) has large coefficients. This makes it more robust in practice than the second-order method in Breuer and Petruccione [21]. Furthermore, we also show that the method has a completely positive property and it can be easily written in a Kraus form. This makes it possible to implement such a simulation method on a quantum computer as well. This is done by unraveling the Lindblad dynamics to a stochastic Schrödinger equation [21], followed by a stochastic expansion [25].

Another practical aspect of our approach is efficient optimization. With the Kraus representation of our numerical approximation, the calculation of the gradient is streamlined. This allows us to apply the Levenberg- Marquardt algorithm, which has very rapid convergence [26, 27]. With mild assumptions on the fitting error, we will show how the approximation error from the numerical simulation and the statistical error from the measurements affect the performance of the parameter identification.

The rest of the paper is organized as follows: We present a general learning framework in the form of a nonlinear least squares problem in Section 3.1, and explain the difference and connections to existing works in Section 3.2. To emphasize the algorithmic details, we present the methods without reference to specific open quantum systems, also with the hope that the methods can be applied to a broader class of problems. Then in Section 3.3 and Section 3.4, we present the specific simulation method and how it is integrated with the optimization procedure. In Section 3.5 and Section 4, we present some error analysis and results from some numerical experiments, where we detail the specific applications of open quantum systems.

2 Notations and Terminologies

Throughout this paper, the Euclidean norm is denoted by $\|\cdot\|$, i.e. for $\boldsymbol{v} \in \mathbb{C}^d$, $\|\boldsymbol{v}\| = (\sum_{i=1}^d v_i^2)^{1/2}$. The density operator $\rho \in \mathbb{C}^{d\times d}$ is represented as a semi-positive definite matrix with trace 1, properties that will be expressed as $\rho \succeq 0$ and $\mathrm{tr}(\rho) = 1$. An emphasis will be placed on quantum evolutions that are trace-preserving and completely positive. These properties are formalized in terms of dynamic maps $\mathbb{C}^{d\times d} \to \mathbb{C}^{d\times d}$, e.g. see [28]. In particular, if $\mathrm{tr}(\mathcal{A}(\rho)) = \mathrm{tr}(\rho)$ for all ρ, then $\mathcal{A}$ is said to be trace-preserving. Similarly, $\mathcal{A}$ is said to be positive if $\rho \succeq 0 \Rightarrow \mathcal{A}\rho \succeq 0$. Further, $\mathcal{A}$ is said to be completely positive if $\mathcal{A} \otimes I_{m\times m}$ is a positive map for every $m \geq 1$. A useful representation of trace preserving and completely positive (CPTP) maps is the Kraus form, which expresses $\mathcal{A}$ as follows,

$$\mathcal{A}\rho = \sum_j V_j \rho V_j^\dagger, \quad \text{with} \sum_j V_j^\dagger V_j = I \tag{2}$$

In this paper, τ_m and t_n respectively denote the simulation and measurement times. The corresponding time interval is represented by δt and Δt, respectively, with the ratio denoted by $L = \frac{\Delta t}{\delta t}$, see Fig. 1. In particular, the simulation times are designed to be shorter or equal to the measurement times. The ratio L can be controlled to obtain the desired accuracy.

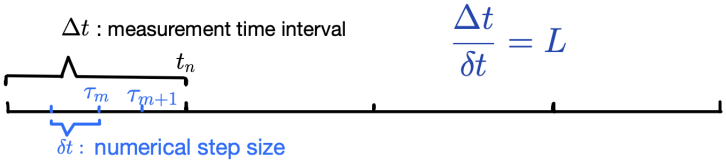

Figure 1: Time steps associated with the simulations and measurements. The measurement time interval Δt is chopped into smaller intervals with δt representing numerical step size.

3 The Learning Framework for Parameter Identification of Lindbladians

3.1 A Least Squares Formulation of the Learning Problem

In practice, the unknown parameters in an open quantum system can appear in both the Hamiltonian and the dissipative terms in the QME (1). To indicate the role of the parameters, we first rewrite Eq. (1) as follows,

$$\begin{aligned} \frac{d}{dt}\rho &= \mathcal{L}_H(\boldsymbol{\theta}_H)\rho + \mathcal{L}_D(\boldsymbol{\theta}_D)\rho, \\ \mathcal{L}_H(\boldsymbol{\theta}_H)\rho = -i[H,\rho], \quad \mathcal{L}_D(\boldsymbol{\theta}_D)\rho &= \sum_{j=1}^{N_V} \left(V_j \rho V_j^\dagger - \frac{1}{2} V_j^\dagger V_j \rho - \frac{1}{2} \rho V_j^\dagger V_j \right). \end{aligned} \tag{3}$$

We can combine the parameters by writing $\boldsymbol{\theta} := (\boldsymbol{\theta}_H, \boldsymbol{\theta}_D)$; $\mathcal{L}(\boldsymbol{\theta}) := \mathcal{L}_H(\boldsymbol{\theta}_H) + \mathcal{L}_D(\boldsymbol{\theta}_D)$ will be called the *Lindbladian*. We refer to the two sets of parameters as Hamiltonian and dissipative parameters, respectively. When $\mathcal{L}_D = 0$, the problem is reduced to a Hamiltonian learning problem.

We assume that the measurement data is a time series $y_{k,n}, n = 1, 2, \cdots, N_T$, which correspond to observables $A^{(k)}, k = 1, \cdots, N_O$, at time instances $t_n = n\Delta t, \quad 0 \leq n \leq N_T$,

$$y_{k,n} := \mathrm{tr}\left(A^{(k)} \rho(t_n; \boldsymbol{\theta})\right). \tag{4}$$

We have set $t_0 = 0$, and for simplicity we assume a uniform time-lapse Δt between experiments, although the extension to general distributions of measurement times is straightforward. Also indicated in Eq. (4) is the dependence of the solution of (3) on the parameters, i.e., $\rho(t_n;\boldsymbol{\theta}) = e^{t_n\mathcal{L}(\boldsymbol{\theta})}\rho(0)$.

We now formulate the learning problem based on the consistency in Eq. (4), as a least squares problem,

$$\min_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}), \tag{5}$$

where the objective function is given by,

$$\phi(\boldsymbol{\theta}) = \frac{1}{2N_O N_T} \sum_{\substack{k=1,\cdots,N_O \\ n=1,\cdots,N_T}} \left| y_{k,n} - y^*_{k,n} \right|^2, \quad y^*_{k,n} := \mathrm{tr}\left(A^{(k)}\rho(t_n;\boldsymbol{\theta}^*)\right), \tag{6}$$

Here, $y^*_{k,n}$ is interpreted as the measurement data with respect to the true, but unknown parameter $\boldsymbol{\theta}^*$; $\boldsymbol{\theta}^*$ is the solution of the least squares problem, i.e., $\boldsymbol{\theta}^* = \mathrm{argmin}\phi(\boldsymbol{\theta})$. Due to the typical nonlinear dependence of $\rho(t_n;\boldsymbol{\theta})$ on $\boldsymbol{\theta}$, the optimization problem is a nonlinear least squares problem [29].

An alternative viewpoint is a maximum likelihood estimate (MLE) from parameter estimation methods for dynamical systems [30]. Namely, the measurement outcome $y_{k,n}$ comes with a Gaussian noise $\epsilon_{k,n}$,

$$y_{k,n} = \mathrm{tr}\big(A^{(k)}\rho(t_n;\boldsymbol{\theta})\big) + \epsilon_{k,n}.$$

Then Eq. (6) is the log-likelihood function.

3.2 Related Works

The QME (1) is an important alternative to study electron transport properties, which are important in material science [31, 32]. Attempts have been made to integrate the model with existing parallel code [33] to simulate systems with many electrons. These simulation algorithms can be regarded as a bottom-up approach, where the bath operators are assumed in advance. Another increasingly important application is quantum computing, where the quantum circuits are subject to environmental noise, and many quantum error mitigation schemes require the knowledge of an error model, i.e., the channel interactions and strength [34, 35, 36].

Bairey et al. [18] proposed to learn the coefficients in the QME (1) by acting the steady-state equation on a collection of observables. These equations can thus be rewritten in terms of the expectations by applying the adjoint of the Lindbladians to the observables, leading to equations of Ehrenfest form. For example, by letting $\langle A\rangle(t) := \mathrm{tr}(A\rho(t))$, one can derive from Eq. (1),

$$\frac{d}{dt}\langle A\rangle(t) = -i\langle[A,H]\rangle(t) + \sum_{j=1}^{N_V}\left(\langle V_j^\dagger A V_j\rangle(t) - \frac{1}{2}\langle A V_j^\dagger V_j\rangle(t) - \frac{1}{2}\langle V_j^\dagger V_j A\rangle(t)\right). \tag{7}$$

The operators on the right-hand side can be expressed on a known basis with unknown parameters, which reformulate the equation in terms of measurement data and parameter values. At a steady state, the time derivative drops out, and the approach by Bairey et al. [18] yields a system of equations, usually over-determined, for the unknown parameters. Franca et al. [19] extended this framework to dynamics problems, with the time derivatives estimated from polynomial approximations. As alluded to in the previous section, these two

approaches can be summarized as *equation-based:* in that the objective function is set up so that the equation (7) holds the best it can. An important practical issue is that the time interval for the measurements, denoted in this article by Δt, might be large, in which case, the accuracy of the inference is limited by the approximation of the time derivatives. If there is no means to decrease the sample time interval, we face inherent inaccuracies in the learning method. More importantly, the measurement data in practical applications always contain noise, leading to a subtle challenge: For the numerical differentiation procedure to be accurate, Δt must be sufficiently small. However, a small Δt can amplify the effect of measurement noise. This difficulty has been identified and analyzed in other applications [37, 38]. The approach in [19] mitigated this issue by choosing the measurement times based on the Chebyshev measure, which on the other hand, requires an upper bound on the time interval $[0, t_{\max}]$. Another challenge is that the right-hand side of Eq. (7) may involve many expectations that have to be obtained from measurements. It is also worthwhile to mention that Boulant et al. [39] also proposed numerical algorithms for reconstructing Lindblad operators and highlighted the importance of the CP property. But their method to ensure this property is quite involved.

The more recent work [40] aims at reconstructing Lindbladians based on measurements at multiple times with an optimization problem. Their method is based on experimental measurements and the least squares problem is formulated in terms of the discrete probability induced by the observables. Numerical simulations are only used to determine a stopping criterion. Compared to this experiment-based approach, the simulation-assisted approach only works with the original set of measurement data. More importantly, the numerical simulation provides the gradient of the objective function, which can substantially speed up the optimization process. Another related approach to Lindbladian learning is the learning algorithms from Gibbs states [12, 14], since the Gibbs state could be reached by certain Lindblad dynamics.

While the equation-based methods [18, 19] fundamentally differ from the method presented in this paper, the two approaches can be combined to accelerate the parameter estimation procedure. Indeed, the current problem can be viewed, in the broader context, as parameter estimation of ODEs [30], where the polynomial approximation of derivatives is known as polynomial regression. It has been used as a preparation of a two-step estimation procedure [41, 42]. The parameters obtained from the linear regression, e.g., in the approach of Franca et al. [19] can be used as an initial guess for a trajectory-based approach.

3.3 Computing the Objective Function using Direct Simulations of the QME

Solving the optimization problem in (5) requires access to the expectation of $A^{(k)}$ for an approximate parameter $\boldsymbol{\theta}$, and such data are unavailable from experiments. This is treated by an efficient numerical algorithm for solving the QME (3). Here we outline a simple derivation of numerical methods, which can be implemented on either classical or quantum computers.

In a nutshell, to approximate the solution operator $e^{\mathcal{L}t}$ of the QME (1), a simulation-assisted algorithm uses an approximation $\mathcal{M}_{\delta t}(t)$ and minimizes the difference between $e^{\mathcal{L}^\dagger(t_n)}A$ and $\mathcal{M}^\dagger_{\delta t}(t_n)A$, by taking the trace with $\rho(0)$. Another advantage of this approach is that we no longer have to measure the terms on the right-hand side, as was done in [18, 19]. Thus the number of measurements can be significantly reduced.

The QME encodes a trace-preserving and completely positive (TPCP) map [22, 23],

which in the Kraus form, can be generally written as [28] (Choi-Kraus'theorem),

$$\rho_{n+1} = \mathcal{K}[\rho_n] := \sum_j F_j \rho_n F_j^\dagger, \tag{8}$$

where $\sum_j F_j^\dagger F_j = I$[21]. Therefore, if such a Kraus form associated with the solution $e^{t_n \mathcal{L}(\boldsymbol{\theta})}$ of the QME (3) can be found, then the objective function in Eq. (6) and its gradient can be directly evaluated to carry out the optimization task. Another interesting aspect is that this procedure also places the problem in the general framework of learning a quantum system [3]. Boulant et al. [39] have demonstrated that the CP property increases the overall robustness of the parameter estimation procedure.

Our approach starts by first unraveling the Lindblad equation (3) into the stochastic Schrödinger equation [21],

$$\mathrm{d}\,|\psi(t)\rangle = \left(-iH\,|\psi(t)\rangle - \frac{1}{2}\sum_{j=1}^{N_V} V_j^\dagger V_j\,|\psi(t)\rangle\right)\mathrm{d}t + \sum_{j=1}^{N_V} V_j\,|\psi(t)\rangle\,\mathrm{d}W_{j;t} \tag{9}$$

Here $|\psi(t)\rangle$ represents a stochastic realization of a quantum state $\rho(t)$, in the sense that,

$$\rho(t) = \mathbb{E}\big[\,|\psi(t)\rangle\langle\psi(t)|\,\big]. \tag{10}$$

In the stochastic equation, $W_{j;t}$'s are independent Brownian motions that incorporate the noise from the bath.

Although in practice the trace-preserving property can be ensured by simple scaling, the completely positive property is often destroyed by classical ODE methods [43]. A simple example is the Euler's method, e.g., applied to a simple Lindblad equation: $\frac{d}{dt}\rho = -\frac{1}{2}\{V^\dagger V, \rho\} + V\rho V^\dagger$,

$$\rho_{n+1} = \rho_n - \frac{1}{2}\delta t V V^\dagger \rho_n - \frac{1}{2}\delta t \rho_n V^\dagger V + \delta t V \rho_n V^\dagger.$$

The right hand side is a Kraus form, but not in a diagonal form: It can be written as, $\rho_{n+1} = \sum_{i,j=1}^3 a_{i,j} F_i \rho_n F_j^\dagger$, with $F_1 = I, F_2 = V, F_3 = VV^\dagger$. But the corresponding matrix $(a_{i,j})$ is clearly not positive definite. Consequently, standard ODE methods may not induce a CP map.

The equivalence between the QME (1) and Eq. (10) is the key ingredient for constructing an approximation of the QME that preserves the CP property. As a quick demonstration, we first consider a time discretization of the stochastic Schrödinger equation (9) by the semi-implicit Euler method [25]. Toward this end, we define numerical time steps, $\tau_m = m\delta t, m = 0, 1, \cdots, M$, and an approximate wave function $|\psi_m\rangle$,

$$|\psi(\tau_m)\rangle \approx |\psi_m\rangle\,,\ m \geq 0. \tag{11}$$

It is important to point out that the step size δt is a numerical parameter, and it can be much smaller than the measurement time Δt. In order for the simulation to produce expectations at the same time steps as the experiments, choose δt such that,

$$L = \frac{\Delta t}{\delta t} \in \mathbb{N}, \quad M = LN_T. \tag{12}$$

The semi-implicit Euler method follows a time-marching step and implements an iteration formula for $|\psi_m\rangle$ as follows,

$$|\psi_{m+1}\rangle = |\psi_m\rangle + G\,|\psi_{m+1}\rangle\,\delta t + \sum_{j=1}^{N_V} V_j\,|\psi_m\rangle\,\delta W_{j,m}. \tag{13}$$

The terminology "semi-implicit" comes from the treatment of the noise: It is only sampled from the current time step and if $V_j = 0$, this method is the standard implicit Euler method for an ordinary differential equation. In Eq. (13) $\delta W_{j,m}$'s are independent Gaussian random variables with mean zero and variance δt. In addition, the matrix G is given by,

$$G = -iH - \frac{1}{2}\sum_{j=1}^{N_V} V_j^\dagger V_j. \tag{14}$$

It also appears in the structure-preserving scheme [43].

We can express the method in a compact form

$$|\psi_{m+1}\rangle = (I - G\delta t)^{-1}\left[I + \sum_{j=1}^{N_V} V_j \delta W_{j,m}\right]|\psi_m\rangle\,. \tag{15}$$

At this point, an approximation method for the density operator ρ can easily be obtained. Specifically, let $\rho_m = \mathbb{E}[\,|\psi_m\rangle\langle\psi_m|\,], m \geq 0$; $\rho(\tau_m) \approx \rho_m$. By taking expectations of (15), we arrive at,

$$\rho_{m+1} = (I - G\delta t)^{-1}\rho_m(I - G\delta t)^{-\dagger} + (I - G\delta t)^{-1}\sum_{j=1}^{N_V} V_j\rho_m V_j^\dagger (I - G\delta t)^{-\dagger}\delta t. \tag{16}$$

Here we have used $\mathbb{E}[\delta W_{j,m}] = 0$ and $\mathbb{E}[\delta W_{j,m}\delta W_{k,m}] = \delta t\delta_{j,k}$.

Clearly, this can be written in the Kraus form (8). The corresponding Kraus operators are given by,

$$F_0 = (I - G\delta t)^{-1}, \tag{17}$$

$$F_j = (I - G\delta t)^{-1}V_j\sqrt{\delta t}, j = 1, \cdots, N_V. \tag{18}$$

This Kraus form from the semi-implicit Euler method is summarized in Lemma 1.

Lemma 1. *The semi-implicit Euler method induces a Kraus form, i.e.,* $\rho_{m+1} = \mathcal{K}[\rho_m] = \sum_j F_j\rho_m F_j^\dagger$ *where* $\sum_j F_j^\dagger F_j = I + \mathcal{O}(\delta t^2)$. *In addition,*

$$e^{\delta t\mathcal{L}(\boldsymbol{\theta})}\rho - \mathcal{K}\rho = \mathcal{O}(\delta t^2). \tag{19}$$

The approximation property can be verified by direct expansions of the Kraus operators in Eq. (17).

In practice, such first-order methods have limited accuracy. To ensure better performance, we consider the second-order implicit approximation method for stochastic differential equations [25, Chapter 15]. When applied to Eq. (9), the method can be written as,

$$\begin{aligned}|\psi_{m+1}\rangle = &(I - \frac{1}{2}G\delta t)^{-1}(I + \frac{1}{2}G\delta t)\,|\psi_m\rangle + \sum_{j=1}^{N_V}(I - \frac{1}{2}G\delta t)^{-1}(V_j + \frac{1}{2}V_jG\delta t)\,|\psi_m\rangle\,\delta\hat{W}_j \\ &+ \frac{1}{2}(I - \frac{1}{2}G\delta t)^{-1}\sum_{j_1,j_2=1}^{N_V} V_{j_2}V_{j_1}\,|\psi_m\rangle\left(\delta\hat{W}_{j_1,m}\delta\hat{W}_{j_2,m} + U_{j_1,j_2,m}\right).\end{aligned} \tag{20}$$

In this expression, $\delta\hat{W}$ is an approximation of an increment of the Brownian motion. In addition, U_{j_1,j_2} are independent two-point distribution, defined as,

$$\begin{aligned} &P(U_{j_1,j_2} = \pm\delta t) = \frac{1}{2}, \forall j_2 = 1, \cdots, j_1 - 1, \\ &U_{j_1,j_1} = -\delta t, \\ &U_{j_1,j_2} = -U_{j_2,j_1}, \text{ for } j_2 = j_1 + 1, \cdots, N_V . j_1 = 1, \cdots, N_V, \end{aligned} \tag{21}$$

We now state the approximation property of this method.

Lemma 2. *The implicit second-order approximation induces a Kraus form, i.e.* $\rho_{m+1} = \sum_j F_j \rho_m F_j^\dagger$, *where* $\sum_j F_j F_j^\dagger = I + \mathcal{O}(\delta t^3)$. *In addition, the one-step error is given by,*

$$e^{\delta t \mathcal{L}} \rho = \sum_j F_j \rho F_j^\dagger + \mathcal{O}(\delta t^3). \tag{22}$$

Proof. Similar to Lemma 1, we define $\rho_m = \mathbb{E}[|\psi_m\rangle \langle\psi_m|]$. Following the second-order implicit scheme (20), each iteration of the density operator is as follows,

$$\begin{aligned} \rho_{m+1} &= (I - \frac{1}{2}G\delta t)^{-1}(I + \frac{1}{2}G\delta t)\rho_m(I + \frac{1}{2}G\delta t)(I - \frac{1}{2}G\delta t)^{-1} \\ &+ (I - \frac{1}{2}G\delta t)^{-1} \sum_{j=1}^{N_V} V_j (I + \frac{1}{2}G\delta t)\rho_m (I + \frac{1}{2}G\delta t) V_j^\dagger (I - \frac{1}{2}G\delta t)^{-1}\delta t \\ &+ \frac{1}{2}(I - \frac{1}{2}G\delta t)^{-1} \sum_{j_1,j_2=1}^{N_V} V_{j_1} V_{j_2} \rho_m V_{j_2}^\dagger V_{j_1}^\dagger (I - \frac{1}{2}G\delta t)^{-1}(\delta t)^2 \\ &= \sum_{j=0}^{N_V^2+N_V} F_j \rho_m F_j^\dagger =: \mathcal{K}(\rho_m) \end{aligned} \tag{23}$$

where the Kraus operators are given by,

$$\begin{aligned} F_0 &= (I - \frac{1}{2}G\delta t)^{-1}(I + \frac{1}{2}G\delta t), \\ F_j &= (I - \frac{1}{2}G\delta t)^{-1} V_j (I + \frac{1}{2}G\delta t)\sqrt{\delta t}, j = 1, \cdots, N_V, \\ F_{j_1+N_V j_2} &= \frac{1}{\sqrt{2}}(I - \frac{1}{2}G\delta t)^{-1} V_{j_1} V_{j_2} \delta t, \quad j_1, j_2 = 1, \cdots, N_V. \end{aligned} \tag{24}$$

Notice that the one-step error is of the third order, so we can simplify the iteration formula without changing the error order by letting

$$F_{j_1+N_V j_2} = \frac{1}{\sqrt{2}} V_{j_1} V_{j_2} \delta t, \quad j_1, j_2 = 1, \cdots, N_V. \tag{25}$$

This will simplify the calculation of the gradient. The rest of the proof is given in Appendix A.2.

□

We chose an implicit method due to its numerical stability. For example, one can show that the spectral radius of each Kraus operator has an upper bound that is independent of H. This can be seen from the observation that the real part of the matrix G is positive definite. Therefore, the spectral radius of F_0 is less or equal to 1. For $0 < j \leq N_V$, F_j

has a spectral radius less or equal to the spectral radius of $V_j\sqrt{\delta t}$. The matrix inverse of $I - \frac{1}{2}G\delta t$ will certainly complicate the numerical implementation. One robust approach is to solve the associated linear system of equations by bi-conjugate gradient method (Bi-CGSTAB) [44], which involves matrix-vector multiplication and orthogonalization. On the other hand, if H does not involve large eigenvalues, an explicit method can be used to replace Eqs. (13) and (20) to simplify the implement, e.g., using the second-order Itô-Taylor expansion [25].

3.4 Gradient Evaluations for Gradient-based Optimization

The problem of finding the optima of $\phi(\boldsymbol{\theta})$ can be seen as the nonlinear least squares problem

$$\phi(\boldsymbol{\theta}) = \frac{1}{2N_O N_T} \sum_{\substack{k=1,\cdots,N_O \\ n=1,\cdots,N_T}} |r_{k,n}(\boldsymbol{\theta})|^2 = \frac{1}{2N_O N_T} R(\boldsymbol{\theta})^T R(\boldsymbol{\theta}), \tag{26}$$

where we used the textbook notations [29]: $r_{k,n} = y_{k,n} - y^*_{k,n}$ is the residual error and $R(\boldsymbol{\theta}) = (r_{k,n})_{k,n}$ will be referred to as *the residual* vector. We are interested in the scenario when the dimension of the residual is larger than the number of parameters, i.e., the over-determined regime.

One practical iteration method for solving the least squares problem is the Levenberg-Marquardt (LM) algorithm, which, starting with an initial guess, $\boldsymbol{\theta}^{(0)}$, involves updating the parameters iteratively,

$$\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\theta}^{(k)} = -\left(\nu_k I + R'(\boldsymbol{\theta}^{(k)})^T R'(\boldsymbol{\theta}^{(k)})\right)^{-1} R'(\boldsymbol{\theta}^{(k)})^T R(\boldsymbol{\theta}^{(k)}). \tag{27}$$

It can be seen as a combination of the Gauss-Newton method and the gradient descent algorithm. In Eq. (27), R' denotes the gradient of the residual with respect to the parameters. The parameter ν_k serves as a regularization, and in practice, it can be chosen to be proportional to the norm of the residual error.

The convergence of the LM method has been established under very mild conditions. Here we follow [26, 27] and pose the following local condition: There exists a constant C, such that,

$$\|R(\boldsymbol{\theta})\| \geq C\|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|, \tag{28}$$

for all $\boldsymbol{\theta}$ in a neighborhood of $\boldsymbol{\theta}^*$.

Theorem 1. *[26, Theorem 2.1] Assume that Eq. (28) holds and R' is Lipschitz continuous. Let the parameter ν_k be*

$$\nu_k = \left\|R(\boldsymbol{\theta}^{(k)})\right\|^2.$$

If the initial guess $\boldsymbol{\theta}^{(0)}$ is sufficiently close to $\boldsymbol{\theta}^*$, then there exists a constant, such that the iterations from (27) satisfy quadratic convergence,

$$\left\|\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\theta}^*\right\| \leq C\left\|\boldsymbol{\theta}^{(k)} - \boldsymbol{\theta}^*\right\|^2. \tag{29}$$

The convergence speed is faster than the super-linear convergence proved in [29].

The quadratic convergence property makes the LM method (27) an extremely useful algorithm. Global convergence has also been analyzed in [26, 27]. This is often accomplished by using a line search algorithm.

Remark 1. *The LM algorithm's fast convergence comes with the cost of computing the inverse of a matrix in (27). One alternative, which is particularly useful for large scale problems, is the gradient descent algorithm,*

$$\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\theta}^{(k)} = -\alpha_k R'(\boldsymbol{\theta}^{(k)})^T R(\boldsymbol{\theta}^{(k)}),$$

where α_k is the learning rate. Such an algorithm can often achieve linear convergence in a neighborhood of the minimizer [45].

In parameter estimation problems, the condition in Eq. (28) is viewed as a local identifiability condition [30]. Meanwhile, the Lipschitz condition can be verified by assuming that the first order and second derivatives of $\mathcal{L}(\boldsymbol{\theta})$ are bounded, which holds trivially if the Lindbladian has a linear dependence on the parameters. To elaborate on this, a partial derivative of the residual error is given by,

$$\frac{\partial}{\partial\theta_\alpha} r_{k,n}(\boldsymbol{\theta}) = \mathrm{tr}(A^{(k)}\partial_{\theta_\alpha}\rho(t_n;\boldsymbol{\theta})). \tag{30}$$

This leads us to consider $\Gamma_\alpha(t,\boldsymbol{\theta}) := \partial_{\theta_\alpha}\rho(t;\boldsymbol{\theta})$, which from Eq. (3), follows the differential equation,

$$\frac{d}{dt}\Gamma_\alpha(t,\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta})\Gamma_\alpha(t,\boldsymbol{\theta}) + \frac{\partial}{\partial\theta_\alpha}\mathcal{L}(\boldsymbol{\theta})\rho(t,\boldsymbol{\theta}).$$

Using the contraction property of $e^{t\mathcal{L}}$ [46], we find that,

$$\|\Gamma_\alpha(t,\boldsymbol{\theta})\| \leq t\|\partial_{\theta_\alpha}\mathcal{L}(\boldsymbol{\theta})\|.$$

To examine the Lipschitz continuity of the partial derivatives, we define,

$$\chi_{\alpha,\beta} := \partial_{\theta_\beta}\Gamma_\alpha(t,\boldsymbol{\theta}),$$

which follows the equation,

$$\frac{d}{dt}\chi_{\alpha,\beta}(t,\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta})\chi_{\alpha,\beta}(t,\boldsymbol{\theta}) + \frac{\partial^2}{\partial\theta_\beta\partial\theta_\alpha}\mathcal{L}(\boldsymbol{\theta})\rho(t,\boldsymbol{\theta}) + \frac{\partial}{\partial\theta_\alpha}\mathcal{L}(\boldsymbol{\theta})\Gamma_\beta(t,\boldsymbol{\theta}) + \frac{\partial}{\partial\theta_\beta}\mathcal{L}(\boldsymbol{\theta})\Gamma_\alpha(t,\boldsymbol{\theta}),$$

and a simple bound follows,

$$\|\chi_{\alpha,\beta}(t,\boldsymbol{\theta})\| \leq t\left\|\frac{\partial^2}{\partial\theta_\alpha\partial\theta_\beta}\mathcal{L}(\boldsymbol{\theta})\right\| + \frac{t^2}{2}(2\|\partial_{\theta_\alpha}\mathcal{L}(\boldsymbol{\theta})\| + \left\|\partial_{\theta_\beta}\mathcal{L}(\boldsymbol{\theta})\right\|).$$

As a result, if the first and second order derivatives of $\mathcal{L}(\boldsymbol{\theta})$ are bounded, R' fulfills the Lipschitz condition.

Meanwhile, in our learning task, the residual function is subject to approximation error. To incorporate these errors, we can follow the proof in [26], where the search direction at each step of the LM algorithm involves a linear least-square (LS) problem. Based on the classical sensitivity analysis for LS [47], an $\mathcal{O}(\epsilon)$ perturbation of $R(\boldsymbol{\theta})$ and $R'(\boldsymbol{\theta})$ will introduce an $\mathcal{O}(\epsilon)$ error in the iteration formula. Therefore we have

Proposition 1. *Let $\epsilon > 0$. Suppose that the residual vector R and R' is subject to an ϵ perturbation,*

$$R \to R + \Delta R,$$

with $\|\Delta R\| \leq \epsilon$ *and* $\|\Delta R'\| \leq \epsilon$*, for all* $\boldsymbol{\theta}$ *in a neighborhood of* $\boldsymbol{\theta}^*$*. Then the LM iterations, under the same conditions on* R *as in the previous theorem, exhibit an approximate quadratic convergence,*

$$\left\|\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\theta}^*\right\| \leq C\left\|\boldsymbol{\theta}^{(k)} - \boldsymbol{\theta}^*\right\|^2 + \epsilon. \tag{31}$$

where C *is independent of* ϵ*.*

The LM algorithm in Eq. (27) requires access to the gradient of the residual function. To facilitate the calculation of the gradient, we first take the derivative of the Kraus form,

$$(\partial_\theta \mathcal{K})[\rho] = \sum_j \partial_\theta F_j \rho F_j^\dagger + F_j \rho \partial_\theta F_j^\dagger \tag{32}$$

Here θ refers to one parameter in $\boldsymbol{\theta}$. The entire gradient can be computed by visiting all the components in $\boldsymbol{\theta}$. For clarity, we express the results as an inner product in the space of Hermitian matrices. Namely, for any Hermitian matrices A and B, we define,

$$\langle A, B\rangle := \mathrm{tr}(AB). \tag{33}$$

Now we show how the gradient of the objective function can be computed from a back propagation procedure.

Lemma 3. *Assume that the iteration* $\rho_m \to \rho_{m+1}$ *follows a Kraus form,*

$$\rho_{m+1} = \mathcal{K}\rho_m := \sum_j F_j \rho_m F_j^\dagger. \tag{34}$$

Then, for any Hermitian operator A*, the following identify holds,*

$$\langle A, \rho_{m+1}\rangle = \langle \mathcal{K}^* A, \rho_m\rangle, \tag{35}$$

where, $\mathcal{K}^*$ *stands for the adjoint of* $\mathcal{K}$*. Namely,*

$$\mathcal{K}^*[A] := \sum_j F_j^\dagger A F_j. \tag{36}$$

Proof. By the cyclic property of the trace operator, we have

$$\mathrm{tr}(A\rho_{m+1}) = \sum_j \mathrm{tr}\Big(AF_j\rho_m F_j^\dagger\Big) = \sum_j \mathrm{tr}\Big(F_j^\dagger A F_j \rho_m\Big) = \mathrm{tr}(\mathcal{K}^*[A]\rho_m)$$

□

In light of this Lemma, we have,

Theorem 2. *When the density operator is simulated by the form of* $\rho_{m+1} = \mathcal{K}[\rho_m] := \sum_j F_j \rho_m F_j^\dagger$ *for each simulation time step* δt*, the explicit form of the derivative of the objective function* $\phi(\boldsymbol{\theta})$ *can be written as*

$$\partial_{\theta_\alpha}\phi(\boldsymbol{\theta}) = \frac{1}{N_O N_T}\sum_{k=1}^{N_O}\sum_{n=1}^{N_T}\sum_{l=1}^{nL}\Big(y_{k,n} - y_{k,n}^*\Big)\Big\langle \partial_{\theta_\alpha}\mathcal{K}^* A_{nL-l}^{(k)},\ \rho_{nL-l}\Big\rangle, \tag{37}$$

where $A_{nL-l}^{(k)} := \left(\mathcal{K}^*\right)^{l-1}[A^{(k)}]$ *can be viewed as a back-propagated operator.*

Proof. The detailed proof can be found in Appendix A.1. □

By using Theorem 2, the explicit expression of $\partial_{\theta_\alpha}\phi(\boldsymbol{\theta})$ can be obtained once we know the derivative of F_j. For instance, for the first-order semi-implicit Euler method,

$$\begin{aligned}\partial_\theta F_0 &= F_0\partial_\theta G F_0 \delta t\\ \partial_\theta F_j &= \partial_\theta F_0 V_j\sqrt{\delta t} + F_0\partial_\theta V_j\sqrt{\delta t}, j = 1,\cdots,N_V\end{aligned} \tag{38}$$

Similarly, for the second-order implicit approximation (23), we have,

$$\begin{aligned}\partial_\theta F_0 &= \frac{1}{2}\delta t(I-\frac{1}{2}G\delta t)^{-1}\partial_\theta G(F_0+I)\\ \partial_\theta F_j &= \frac{1}{2}\delta t(I-\frac{1}{2}G\delta t)^{-1}\partial_\theta G F_j + \sqrt{\delta t}(I-\frac{1}{2}G\delta t)^{-1}\partial_\theta V_j(I+\frac{1}{2}G\delta t)\\ &+\frac{1}{2}\delta t^{3/2}(I-\frac{1}{2}G\delta t)^{-1}V_j\partial_\theta G,\quad j=1,\cdots,N_V\\ \partial_\theta F_{j_1+N_V j_2} &= \frac{\delta t}{\sqrt{2}}(\partial_\theta V_{j_1}V_{j_2}+V_{j_1}\partial_\theta V_{j_2}),\quad j_1,j_2=1,\cdots,N_V\end{aligned} \tag{39}$$

We now discuss the time complexity of the overall learning algorithm. Notice that simulating Lindblad dynamics for the time duration $[0,T]$, due to the second order accuracy, incurs complexity that is proportional to the number of time steps, i.e., $LN_T = \mathcal{O}\left(\epsilon^{-1/2}T^{3/2}\right)$. Meanwhile, the quadratic convergence property of the Levenberg-Marquardt algorithm implies that the number of iteration steps is of the order $\mathcal{O}(\log\log\epsilon^{-1})$ such that the optimization error is within ϵ. In addition, with direct calculation, we find that the time complexity of calculating the objective function value and its gradient is $\mathcal{O}(\epsilon^{-1/2}T^{3/2}N_V^2N_{MM})$ and $\mathcal{O}(N_\theta N_O\epsilon^{-1/2}T^{3/2}N_V^2N_{MM})$, respectively, where N_{MM} denotes the complexity associated with a matrix multiplication, e.g., $F_j\rho F_j^\dagger$. Thus the overall complexity is $\tilde{\mathcal{O}}(N_\theta N_O\epsilon^{-1/2}T^{3/2}N_V^2N_{MM})$ where $\tilde{\mathcal{O}}$ ignores logarithmic factors.

Theorem 3. *Given the data sets $\{y^*_{k,n}, k=1,\cdots,N_O, n=1,\cdots,N_T\}$ representing measurements with respect to observables $A^{(k)}$ and time instance t_n until $T=t_{N_T}$. Under the assumption in Theorem 1 and given the observation data and precision ϵ, the learning algorithm yields parameters $\boldsymbol{\theta}$ within ϵ precision and it involves $\tilde{\mathcal{O}}(N_\theta N_O\epsilon^{-1/2}T^{3/2}N_V^2N_{MM})$ arithmetic operations. For a specific n-qubit open quantum system described by Lindblad master equation, where the Hamiltonian H is k-local and the dissipation part $\mathcal{L}$ only contains single qubit terms, the overall complexity of the learning algorithm based on one and two-qubit Pauli observables is*

$$\tilde{\mathcal{O}}\left(\epsilon^{-1/2}n^5T^{3/2}N_{MM}\right).$$

Implicit in the final bound in the above theorem is a k-dependent prefactor, which depends on the structure of the locality terms. The dependence of the bound on n comes from the fact that for this specific quantum system, we have $N_O=\mathcal{O}(n^2)$, $N_\theta=\mathcal{O}(n)$, and $N_V=\mathcal{O}(n)$.

3.5 Quantifying the Error in the Parameter Identification

Similar to modern machine learning problems, there are mainly three sources of error in a quantum learning problem. Specifically, as can be seen from the objective function Eq. (6),

1. The estimated values, $\mathrm{tr}\big(A^{(k)}\rho(t_n;\boldsymbol{\theta})\big)$ are replaced with $\mathrm{tr}\big(A^{(k)}\rho_{nL}(\boldsymbol{\theta})\big)$ (Recall that $\Delta t = L\delta t$, and $\rho(t_n,\boldsymbol{\theta}) = \rho(\tau_{Ln},\boldsymbol{\theta}) \approx \rho_{Ln}(\boldsymbol{\theta})$). This can be regarded as a function approximation error. To include this error, we define,

$$\widehat{r}_{k,n}(\boldsymbol{\theta}) = \mathrm{tr}\big(A^{(k)}\rho_{nL}(\boldsymbol{\theta})\big) - y^*_{k,n}. \tag{40}$$

   Using the second-order method (23), $\widehat{R} = (\widehat{r}_{k,n})_{k,n}$ is a $\mathcal{O}(\delta t^2)$ perturbation of the residual error in the original objective function (6).

2. The data $y^*_{k,n}$, as indicated in (4), come from measuring a set of observables at different time instances, which are subject to statistical error. To clarify this perspective, we express the measurement values as random variables:

$$y^*_{k,n} \approx \hat{y}_{k,n} := \frac{1}{N_S}\sum_{\ell=1}^{N_S} a_{k,n,\ell}. \tag{41}$$

   N_S indicates the number of times the measurements are repeated. The effect of this sampling error can be understood by considering the following objective function,

$$\widetilde{r}_{k,n}(\boldsymbol{\theta}) = \mathrm{tr}\big(A^{(k)}\rho(t_n,\boldsymbol{\theta})\big) - \hat{y}_{k,n}. \tag{42}$$

   Based on Chebyshev's inequality, $\widetilde{R} = (\widetilde{r}_{k,n})_{k,n}$ is a ϵ perturbation of residual error in the original objective function (6) with high probability if $N_S = \mathcal{O}(1/\epsilon^2)$.

3. The optimization problem (5) may lead to a nonlinear system of equations and we may not be able to find the exact solution after a finite number of iterations. This is the optimization error.

The perturbation caused by the numerical approximation (23) is a deterministic perturbation, which from (28), causes a perturbation of the identified parameter.

Theorem 4. *Under the assumption (28), there exists a $\delta_0 > 0$, such that for all $\delta t < \delta_0$, there is a minimizer $\hat{\boldsymbol{\theta}}$ of the least squares problem defined by the residual vector in Eq. (40), such that*

$$\left\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right\| \le C\delta t^2. \tag{43}$$

for some constant C.

We now turn to the measurement error. Since the observables $A^{(k)}$'s are bounded, the measurement noise is bounded, and they can be regarded as a sub-Gaussian random variable. The Hoeffding inequality [48] implies that,

Lemma 4. *There exists $\sigma_{k,n}$, such that the following inequality holds for every $\epsilon > 0$*

$$\mathbb{P}\Big(\Big|\hat{y}_{k,n} - y^*_{k,n}\Big| > \epsilon\Big) \le 2e^{-\frac{N_S\epsilon^2}{\sigma^2_{k,n}}}. \tag{44}$$

Now by using a union bound, we can estimate the probability,

$$\mathbb{P}\Big(\sqrt{\sum_{k,n}\Big|\hat{y}_{k,n} - y^*_{k,n}\Big|^2} > \epsilon\Big) \le 2N_O N_T e^{-\frac{N_S\epsilon^2}{N_O N_T \sigma^2}}.$$

Here we have set $\sigma^2 = \frac{1}{N_O N_T}\sum_{k,n}\sigma^2_{k,n}$.

Theorem 5. *Under the same assumption as in the previous theorem, and further assume that the observation data $y_{k,n}$ are sampled,*

$$N_S = \Omega\left(\frac{\sigma^2 N_O N_T}{\epsilon^2}\log\left(N_O N_T\right)\right), \tag{45}$$

times, then there is a minimizer $\widetilde{\boldsymbol{\theta}}$ of the nonlinear least squares with residual in Eq. (42), such that

$$\left\|\widetilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right\| \leq \epsilon. \tag{46}$$

with high probability.

3.6 Quantum/classical Hybrid Algorithms for Lindblad Simulation

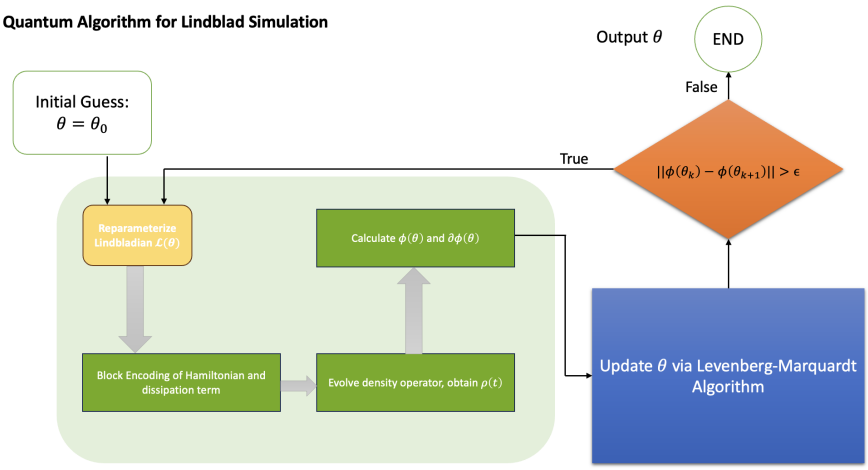

Figure 2: A flowchart describing the parameter estimation algorithm using a quantum algorithm for Lindblad simulation. The simulation scheme is implemented by three steps: 1) Block encoding the Hamiltonian term and jump operators in the Lindbladian with coefficients depending on the input parameter $\boldsymbol{\theta}$; 2) Evolve the density operator $\rho(t)$ to time t; 3) Measure observables $A^{(k)}$ with the density operator $\rho(t)$ and time t_n, and calculate the objective function value $\phi(\theta)$ and its derivatives $\partial\phi(\theta)$. 4) The optimization part is based on Levenberg-Marquardt algorithm on a classical device.

We have thus far discussed a classical approach to simulating the Lindblad equation, which is used to evaluate the objective function $\phi(\boldsymbol{\theta})$ and its gradients as shown in Eq. (37). This method is suitable for quantum systems of moderate size, such as those that can be efficiently treated using large-scale parallel algorithms. Alternatively, these dynamics can be simulated directly on quantum computers. Several such algorithms have already been developed [49, 50, 51, 52]. Due to the many technical elements involved in these algorithms, we will sketch the overall procedure and leave the detailed presentation of these methods in separate works. As illustrated in Fig. 2, the overall procedure can be regarded as a quantum/classical hybrid algorithm, where some approximation of the parameter $\boldsymbol{\theta}$ is fed into the Lindbladian, from which the objective function $\phi(\boldsymbol{\theta})$ is estimated as an expectation. In addition, a gradient estimation algorithm, such as the improved Jordan's algorithm [53], can be used to estimate the gradient of $\phi(\boldsymbol{\theta})$. On the other hand, we run the LM algorithm on a classical device to provide an update of the parameter, which re-enters the quantum Lindblad simulation until it reaches convergence.

4 Numerical Tests

In this section, we present several numerical results to test the effectiveness of our learning algorithm. For the test problem, we consider a quantum system of qubits with dynamics described by Lindblad Eq. (1). In particular, we assume that H is linear combination of $k-$local operators in that each term is acting on at most k qubits. In addition, the jump operators V_j are assumed to be $1-$qubit Paulis. Specifically, we choose the number of qubits $n = 6$. The initial states are fixed as the product state with all spins up. Meanwhile, the observables are chosen to be 1 and 2-local Pauli Strings. The performance of our learning algorithm will be quantified with relative and absolute error with respect to 2-norm. The source code is available at [54].

4.1 Phase Damping Model with Linear Parameter Dependence

We investigate an n-spin system with dephasing and amplitude damping noise on every qubit [34]. The dynamics are described by the QME in Eq. (1), with the parametric form of the Hamiltonian part given by,

$$H = \sum_{j=1}^{n} H_j, \quad H_j = \sum_{\alpha=1}^{3} e_{j,\alpha}\sigma_j^{\alpha} + \sum_{\alpha,\beta=1}^{3} c_{j,\alpha,\beta}\sigma_j^{\alpha}\sigma_{j+1}^{\beta}, \quad j = 1, \cdots, n. \tag{47}$$

σ_j^{α} is the Pauli matrix σ^{α} $(\alpha = x, y, z)$ applied to the jth qubit, and the condition $c_{n,\alpha,\beta} = 0$ models open boundaries.

Meanwhile, the dissipation term is parameterized as follows,

$$\mathcal{L}_D\rho = \lambda_1 \sum_{j=1}^{n} \mathcal{T}[\sigma_j^-]\rho + \lambda_2 \sum_{j=1}^{n} \mathcal{T}[\sigma_j^z]\rho, \tag{48}$$

where the operator $\mathcal{T}[V]$ is defined as

$$\mathcal{T}[V]\rho = V\rho V^{\dagger} - \frac{1}{2}\{V^{\dagger}V, \rho\}. \tag{49}$$

The operators $\sigma_j^{\pm} := 2^{-1}(\sigma_j^x \pm i\sigma_j^y)$ are introduced.

The parameters are expressed as a vector $\boldsymbol{\theta}$, consisting of Hamiltonian coefficients $\boldsymbol{\theta}_H = \{e_{j,\alpha}, c_{j,\alpha,\beta}\}$ and dissipative parameters $\boldsymbol{\theta}_D = (\lambda_1, \lambda_2)$, with 65 unknown parameters in total. To initialize the learning algorithms, these parameters will be generated from a Gaussian distribution, and then held fixed as the exact values of the parameters.

We first test the accuracy of the simulation methods in Section 3.3. As a reference, we conducted direct simulations of the test model in Eqs. (47) and (48) by using very small step size: $\delta t = 10^{-4}$. The measurement with 1-qubit local observable $A := \sigma_2^y$, will serve as the benchmark solution and is denoted by $y_{\text{exact}}(t)$. Fig. 3 displays the evolution of this observable in the interval $t \in [0, 10]$. It contrasts the "exact" solution with results from the semi-implicit Euler method (16) and the second-order implicit method (23) using step size $\delta t = 10^{-2}$, referred to as $y_1(t)$, and $y_2(t)$ respectively. One can observe that the semi-implicit Euler's method yields very good accuracy in the initial period $[0, 0.5]$, but only a qualitatively correct solution in the transient state $[0.5, 4]$. In addition, it tends to smear out the solution in the long run, e.g., in the final period $[6, 10]$. But the second-order implicit method offers much better accuracy in the entire time duration. Due to the high accuracy, the measurement data in the following numerical experiments will be generated using the second-order implicit method with $\delta t = 10^{-2}$.

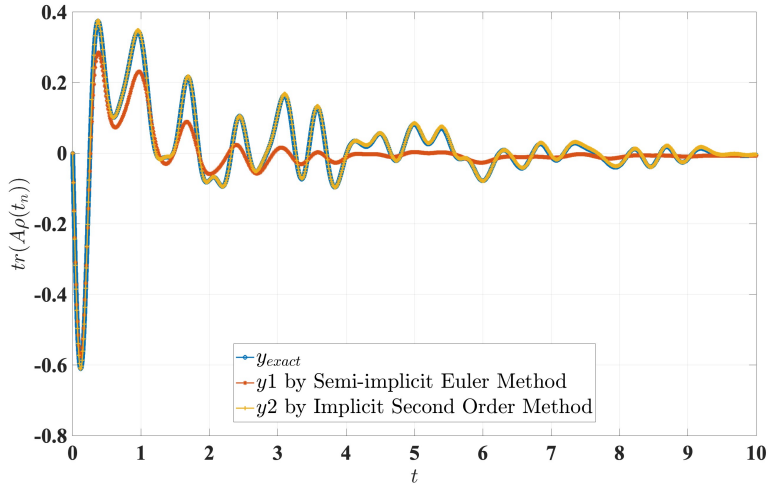

Figure 3: Comparison of the accuracy for the simulation methods in Section 3.3, applied to a 6 -spin chain with dephasing and amplitude damping noise. The density operator ρ is measured with 1-qubit local observables $A := \sigma_2^y$ and the measurement at each time point denoted by $y_n = \mathrm{tr}(A\rho(t_n))$. The blue solid line represents the "exact" solution generated with a very small step size.

We now employ the Lindbladian simulation scheme to implement the optimization algorithms detailed in Section 3.4 . Observations were made at time points $t_n = 0.1, 0.2, \cdots, 1$ with measurement interval $\Delta t = 0.1$ and in the learning algorithm, the underlying Lindblad dynamics were simulated with a smaller step $\delta t = 0.01$ ($L = 10$) . The performance of the algorithms was tested on the basis of all 1-local observables or all 1 and 2-local observables, generated by Pauli matrices. Numerical results in Fig. 4 include the objective function value and the relative error $\|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|/\|\boldsymbol{\theta}^*\|$. Notably, the Levenberg-Marquardt algorithm demonstrated rapid convergence. Furthermore, with all 1 and 2-local observable, the optimization yields slightly faster parameter identification. It demonstrates that an increased number of observables enhances identification.

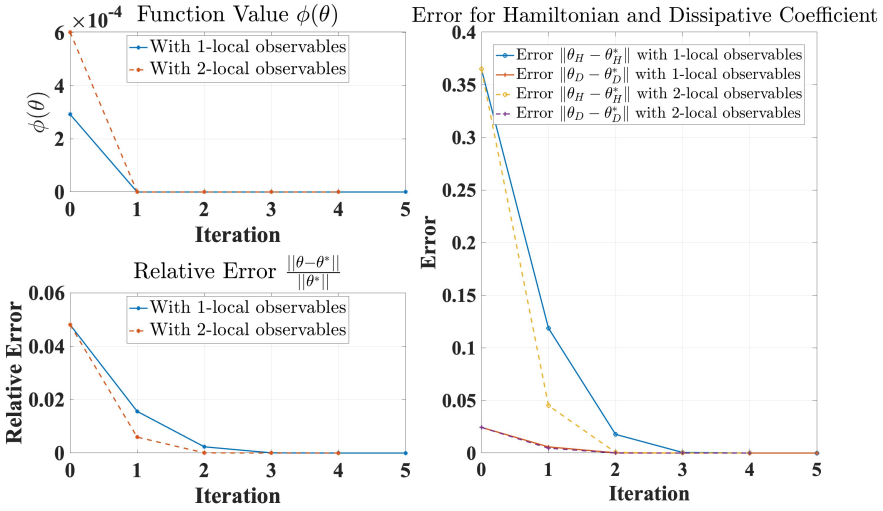

Figure 4: Reconstruction of the Lindbladians in Eqs. (47) and (48) by solving the optimization problem (6). The dataset consists of 1-qubit local observables ($N_O = 19$) and both 1 and 2-local observables($N_O = 154$) at time $t_n = 0.1, 0.2, \cdots, 1$. The numerical simulation time interval is smaller as $\delta t = 0.01$. The initial parameter $\boldsymbol{\theta}^{(0)}$ is randomly selected with initial error $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 0.3658$.

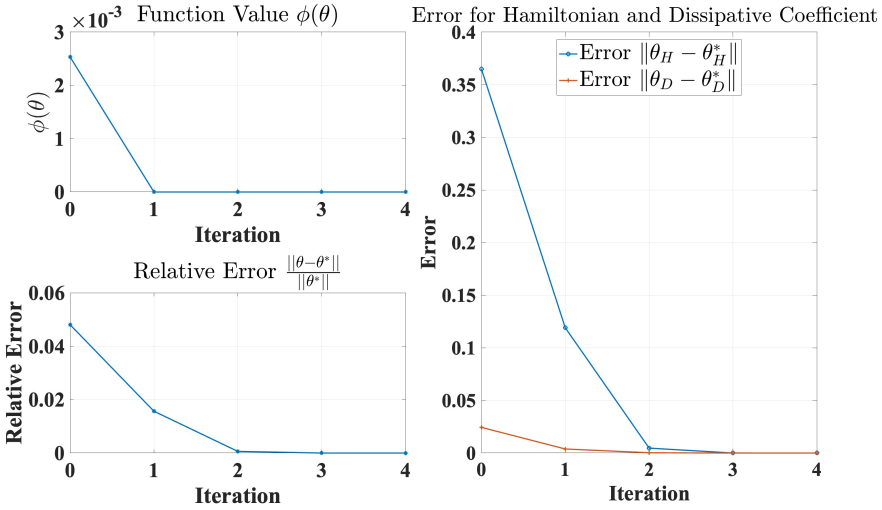

Figure 5: Reconstruction of the Lindbladians in Eqs. (47) and (48) by solving the optimization problem (6). The dataset consists of 1-qubit local observables ($N_O = 19$) at time $t = 0.01, 0.02, \cdots, 1 (N_T = 100)$. The numerical simulation time interval is the same as $\delta t = 0.01$. The initial parameter $\boldsymbol{\theta}^{(0)}$ is the same as Fig. 4.

Extending our analysis, we retained 1-local observables but increased measurement frequency, setting measurement times to $\tau = t = 0.01, 0.02, \cdots, 1$ ($\Delta t = \delta t = 0.01$). Fig. 5 indicates that higher measurement frequency slightly enhances parameter identification. The efficiency of the optimization can again be attributed to the rapid convergence of the LM algorithm.

Inspired by the study in [18], where the Lindbladians are learned from steady states, our last experiment uses observation times at a later stage $t = 4.1, 4.2, \cdots, 5$. A separate numerical test verified that this is when the system is beginning to saturate (see Fig. 3 as well). As shown in Fig. 6, the optimization identified all the parameters. Despite the non-monotonic convergence of $\boldsymbol{\theta}_H$, the last few iterations exhibited rapid convergence.

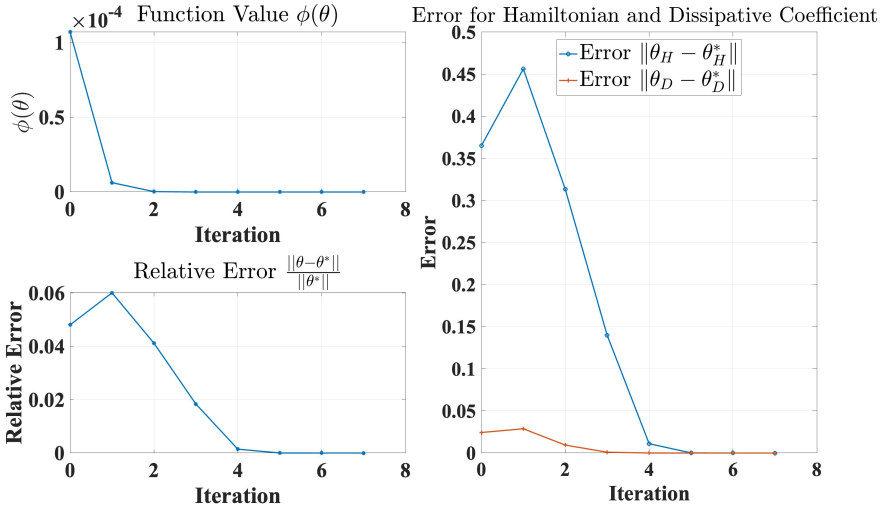

Figure 6: Reconstruction of the Lindbladians in Eqs. (47) and (48) by solving the optimization problem (6). The dataset consists of 1-qubit local observable s($N_O = 19$) at time $t = 4.1, 4.2, \cdots, 5$, ($N_T = 10$). The numerical simulation time interval is smaller as $\delta t = 0.01$. The initial parameter $\boldsymbol{\theta}^{(0)}$ is the same as Fig. 4.

4.2 A Nonlinear Parametric Model

The model in the previous section involves dissipation terms that are linear with respect to the parameters. In this section, we assume that each jump operator in the dissipation term has a linear parametric form, effectively leading to a nonlinear parametric model. This coincides with the example in Bairey et al. [18]. Specifically, the Hamiltonian is of the same linear form as in the previous section while the jump operators are expanded as a linear combination of local Pauli matrices with complex coefficients ,

$$V_j = \sum_{\alpha=1}^{3} d_{j,\alpha}\sigma_j^\alpha = \sum_{\alpha=1}^{3} d_{j,\alpha}^{(1)}\sigma_j^\alpha + i d_{j,\alpha}^{(2)}\sigma_j^\alpha, \quad j = 1, \cdots, N_V. \tag{50}$$

Similarly, the parameters are randomly generated by the Gaussian distribution ,

$$e_{j,\alpha}, c_{j,\alpha,\beta} \sim \mathcal{N}(0,1), \quad \mathrm{Re}(d_{j,\alpha}), \mathrm{Im}(d_{j,\alpha}) \sim \mathcal{N}(0, \frac{1}{2}). \tag{51}$$

These parameters will then be fixed and considered to be exact. Subsequently, they are used to generate the observation data $y_{k,n}^*$ in (4). We have noticed that the parametric form in Eq. (51) is not unique due to the simultaneous appearance of V_j and $V_j^\dagger$ in the Lindbladian. To eliminate the redundancy from the global phase of d_j for each j, we set the imaginary part of $d_{j,1}$ to zeros . The parameter $\boldsymbol{\theta}$ is composed by Hamiltonian part $\boldsymbol{\theta}_H = \{e_{j,\alpha}, c_{j,\alpha,\beta}\}$ as in Eq. (47) and the dissipation part $\boldsymbol{\theta}_D = \{d_{j,\alpha}^{(1)}, d_{j,\alpha}^{(2)}\}$.

The optimization problem (6) with values $y_{k,n}^*$ involved both 1-local and 2-local observables at time $t = 0.1, 0.2, \cdots, 1$ $(\Delta t = 0.1)$. The simulation time interval is much smaller: $\delta t = 0.01$. These experiments (results shown in Fig. 7) indicate that effective parameter identification could be achieved even with the nonlinear parametric model. Moreover, faster convergence was observed when employing all 1 and 2-local observables, which could be possibly caused by a larger constant in the condition (28). To further illustrate the effect of the choice of the observables, we choose 1-local observables, by only keeping σ_j^x and σ_j^y's for each spin, while other configurations of the test remain the same. This modification led to a noticeable increase in the number of iterations to reach a plateau, as demonstrated in Fig. 8. The large optimization error suggests that the limited set of 1-local observables is insufficient for the Lindbladian learning problem.

So far we only considered a small initial guess to show the second-order convergence property of Levenberg-Marquardt algorithm. To illustrate the robustness of our algorithm, we implemented three additional numerical experiments with initial guesses chosen such that the difference between the initial parameter $\boldsymbol{\theta}^{(0)}$ and the minimizer $\boldsymbol{\theta}^*$ is $\mathcal{O}(1)$ for both linear and nonlinear cases. We repeat the numerical experiment in Fig. 5, but using a random initial guess with a much larger error $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 2.3650$. The convergence is within 10 iterations as shown in Fig. 9. Similarly, we repeat the experiment in Fig. 7 and the initial guess has error $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 2.0359$. Our algorithm still converged quite rapidly as depicted in Fig. 10. We now choose $\boldsymbol{\theta}^{(0)}$ with a much larger initial error $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 7.8569$ in the experiment in Fig. 5. As we can observe from Fig. 11, the algorithm still exhibits convergence but the parameter $\boldsymbol{\theta}$ has settled to another minimum. Nevertheless, the objective function is almost zero, indicating an excellent fit to the observation data. In summary, our algorithm demonstrates robust performance even with a considerably larger initial guess. However, the algorithm might end up with another global minimum $\boldsymbol{\theta}$.

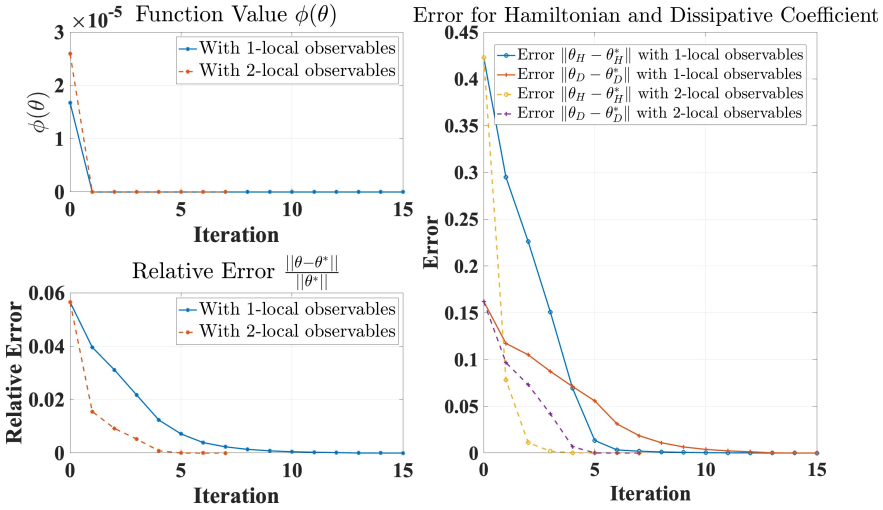

Figure 7: Reconstruction of the Lindbladians in Eqs. (47) and (50) by solving the optimization problem (6). The dataset consists of 1-qubit local observables ($N_O = 19$) and both 1 and 2-local observables($N_O = 154$) at time $t_n = 0.1, 0.2, \cdots, 1$. The numerical simulation time interval is smaller as $\delta t = 0.01$. The initial parameter $\boldsymbol{\theta}^{(0)}$ is randomly selected with initial error $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 0.4529$.

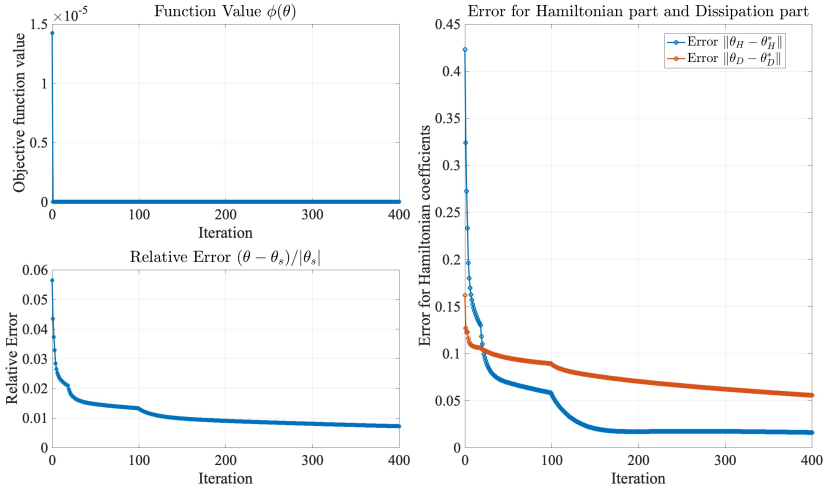

Figure 8: Reconstruction of the Lindbladians in Eqs. (47) and (50) by solving the optimization problem (6). The dataset consists of only 1-qubit local observables σ_j^x, σ_j^y ($N_O = 12$) at time $t_n = 0.1, 0.2, \cdots, 1$ ($N_T = 10$). The numerical simulation time interval is smaller as $\delta t = 0.01$. The initial parameter $\boldsymbol{\theta}^{(0)}$ is the same as the previous test.

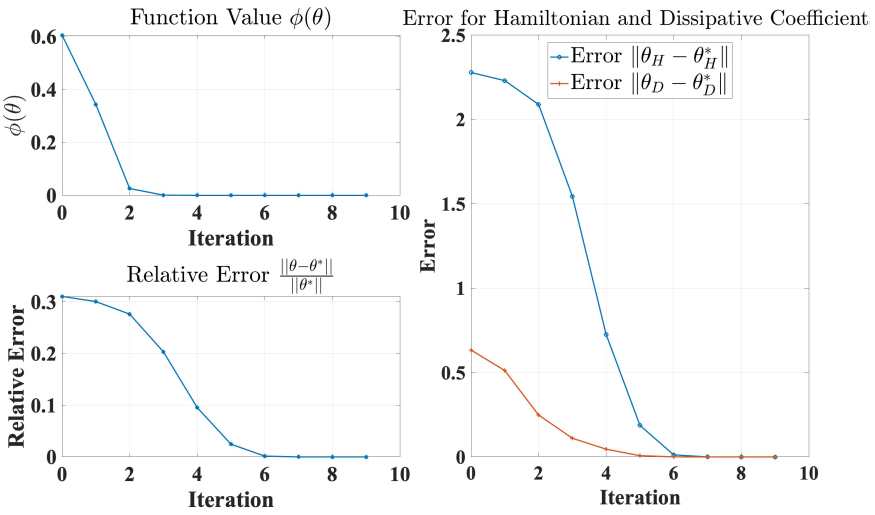

Figure 9: Reconstruction of the Lindbladians of the linear parameter case. The dataset includes 1-qubit local observables ($N_O = 19$) at time $t = 0.01, 0.02, \cdots, 1$ ($N_T = 100$). The numerical simulation time interval is the same as $\delta t = 0.01$. Same setting as Fig. 5. The initial guess is larger, $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 2.3650$.

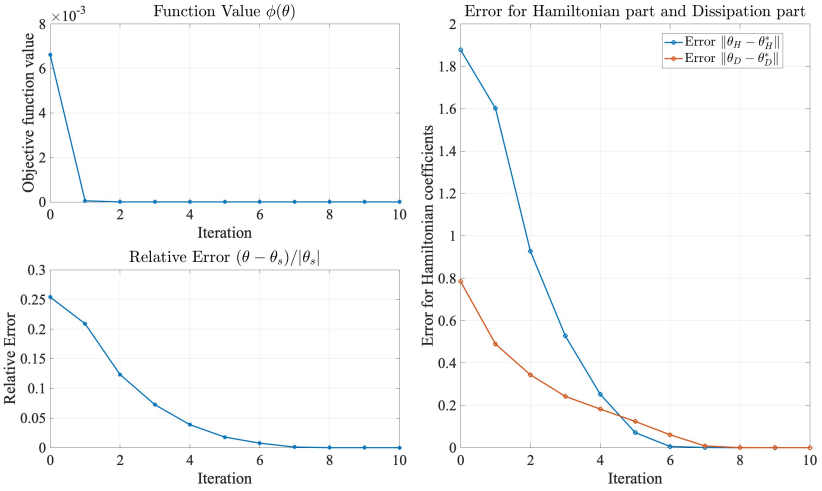

Figure 10: Reconstruction of the Lindbladians of the nonlinear parameter case. The dataset consists of 1-qubit local observables ($N_O = 19$) at time $t = 0.1, 0.2, \cdots, 1$ ($N_T = 10$). The numerical simulation time interval is the same as $\delta t = 0.01$. Same setting as Fig. 7. The initial guess is larger, $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 2.0359$.

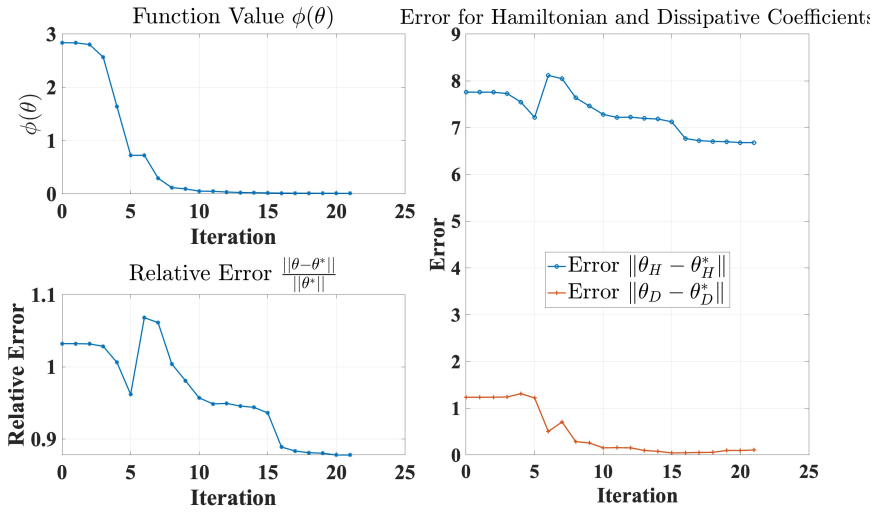

Figure 11: Reconstruction of the Lindbladians of the linear parameter case. The dataset consists of 1-qubit local observables ($N_O = 19$) at time $t = 0.01, 0.02, \cdots, 1$ ($N_T = 100$). The numerical simulation time interval is the same as $\delta t = 0.01$. Same setting as Fig. 5. The initial guess is larger, $\|\boldsymbol{\theta}^{(0)} - \boldsymbol{\theta}^*\| = 7.8569$.

5 Summary and discussions

In this paper, we presented an algorithm to infer parameters in an open quantum system, specifically, in a Lindblad equation. Rather than working with the observation times, we introduce smaller time scales where the solution is obtained by direct numerical simulations. There are two advantages. On one hand, since the objective function is formulated based on trajectories, we no longer need the observables that correspond to the Lindbladian terms that appeared in the approach in [18, 19]. As a result, the number of observables can be much less than the number of jump operators. On the other hand, the algorithm enables flexible control of the accuracy by using smaller step sizes in the numerical simulation.

Acknowledgements

We thank Dr. Di Fang for providing some important references. This work is supported by NSF Grant DMS-1819011 and DMS-2111221.

References

[1] Marcus P da Silva, Olivier Landon-Cardinal, and David Poulin. "Practical characterization of quantum devices without tomography". Physical Review Letters **107**, 210404 (2011).

[2] Jianwei Wang, Stefano Paesani, Raffaele Santagati, Sebastian Knauer, Antonio A Gentile, Nathan Wiebe, Maurangelo Petruzzella, Jeremy L O'brien, John G Rarity, Anthony Laing, et al. "Experimental quantum Hamiltonian learning". Nature Physics **13**, 551–555 (2017).

[3] Hsin-Yuan Huang, Steven T Flammia, and John Preskill. "Foundations for learning from noisy quantum experiments" (2022).

[4] Christopher E Granade, Christopher Ferrie, Nathan Wiebe, and David G Cory. "Robust online Hamiltonian learning". New Journal of Physics **14**, 103013 (2012).

[5] Kenneth Rudinger and Robert Joynt. "Compressed sensing for Hamiltonian reconstruction". Physical Review A **92**, 052322 (2015).

[6] Nathan Wiebe, Christopher Granade, and David G Cory. "Quantum bootstrapping via compressed quantum Hamiltonian learning". New Journal of Physics **17**, 022005 (2015).

[7] Daniel Burgarth and Ashok Ajoy. "Evolution-free Hamiltonian parameter estimation through Zeeman markers". Physical Review Letters **119**, 030402 (2017).

[8] Akira Sone and Paola Cappellaro. "Hamiltonian identifiability assisted by a single-probe measurement". Physical Review A **95**, 022335 (2017).

[9] Yuanlong Wang, Daoyi Dong, Bo Qi, Jun Zhang, Ian R Petersen, and Hidehiro Yonezawa. "A quantum Hamiltonian identification algorithm: Computational complexity and error analysis". IEEE Transactions on Automatic Control **63**, 1388–1403 (2017).

[10] Assaf Zubida, Elad Yitzhaki, Netanel Lindner, and Eyal Bairey. "Optimal short-time measurements for Hamiltonian learning". Bulletin of the American Physical Society**67** (2022).

[11] Hsin-Yuan Huang, Yu Tong, Di Fang, and Yuan Su. "Learning many-body Hamiltonians with Heisenberg-limited scaling". Physical Review Letters **130**, 200403 (2023).

[12] Jeongwan Haah, Robin Kothari, and Ewin Tang. "Optimal learning of quantum hamiltonians from high-temperature gibbs states". In 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS). Pages 135–146. IEEE (2022).

[13] Jeongwan Haah, Aram W Harrow, Zhengfeng Ji, Xiaodi Wu, and Nengkun Yu. "Sample-optimal tomography of quantum states". In Proceedings of the forty-eighth annual ACM symposium on Theory of Computing. Pages 913–925. (2016).

[14] Ainesh Bakshi, Allen Liu, Ankur Moitra, and Ewin Tang. "Learning quantum Hamiltonians at any temperature in polynomial time" (2023).

[15] Hongkang Ni, Haoya Li, and Lexing Ying. "Quantum hamiltonian learning for the fermi-hubbard model". Acta Applicandae Mathematicae **191**, 1–16 (2024).

[16] Haoya Li, Yu Tong, Hongkang Ni, Tuvia Gefen, and Lexing Ying. "Heisenberg-limited hamiltonian learning for interacting bosons" (2023).

[17] M Holzäpfel, T Baumgratz, M Cramer, and Martin B Plenio. "Scalable reconstruction of unitary processes and hamiltonians". Physical Review A **91**, 042129 (2015).

[18] Eyal Bairey, Chu Guo, Dario Poletti, Netanel H Lindner, and Itai Arad. "Learning the dynamics of open quantum systems from their steady states". New Journal of Physics **22**, 032001 (2020).

[19] Daniel Stilck Franca, Liubov A Markovich, VV Dobrovitski, Albert H Werner, and Johannes Borregaard. "Efficient and robust estimation of many-qubit hamiltonians". Nature Communications**15** (2024).

[20] Gabriel O Samach, Ami Greene, Johannes Borregaard, Matthias Christandl, Joseph Barreto, David K Kim, Christopher M McNally, Alexander Melville, Bethany M Niedzielski, Youngkyu Sung, et al. "Lindblad tomography of a superconducting quantum processor". Physical Review Applied **18**, 064056 (2022).

[21] Heinz-Peter Breuer and Francesco Petruccione. "The theory of open quantum systems". Oxford University Press on Demand. (2002).

[22] Goran Lindblad. "On the generators of quantum dynamical semigroups". Communications in Mathematical Physics **48**, 119–130 (1976).

[23] Vittorio Gorini, Andrzej Kossakowski, and Ennackal Chandy George Sudarshan. "Completely positive dynamical semigroups of N-level systems". Journal of Mathematical Physics **17**, 821–825 (1976).

[24] Cambyse Rouzé and Daniel Stilck França. "Learning quantum many-body systems from a few copies" (2021).

[25] P.E. Kloeden and E. Platen. "Numerical Solution of Stochastic Differential Equations". Stochastic Modelling and Applied Probability. Springer Berlin Heidelberg. (2011).

[26] Nobuo Yamashita and Masao Fukushima. "On the rate of convergence of the Levenberg-Marquardt method". In Topics in Numerical Analysis: With Special Emphasis on Nonlinear Problems. Pages 239–249. Springer (2001).

[27] Jin-yan Fan and Ya-xiang Yuan. "On the quadratic convergence of the Levenberg-Marquardt method without nonsingularity assumption". Computing **74**, 23–39 (2005).

[28] John Watrous. "The theory of quantum information". Cambridge university press. (2018).

[29] Carl T Kelley. "Iterative methods for optimization". SIAM. (1999).

[30] Nicolas JB Brunel. "Parameter estimation of ODE's via nonparametric estimators". Electronic Journal of Statistics **2**, 1242–1267 (2008).

[31] Robert Biele and Roberto D'Agosta. "A stochastic approach to open quantum systems". Journal of Physics: Condensed Matter **24**, 273201 (2012).

[32] Massimiliano Di Ventra and Roberto D'Agosta. "Stochastic time-dependent current-density-functional theory". Physical Review Letters **98**, 226403 (2007).

[33] Heiko Appel and Massimiliano Di Ventra. "Stochastic quantum molecular dynamics". Physical Review B **80**, 212303 (2009).

[34] Kristan Temme, Sergey Bravyi, and Jay M Gambetta. "Error mitigation for short-depth quantum circuits". Physical review letters **119**, 180509 (2017).

[35] Suguru Endo, Simon C Benjamin, and Ying Li. "Practical quantum error mitigation for near-future applications". Physical Review X **8**, 031027 (2018).

[36] Zhenyu Cai, Ryan Babbush, Simon C Benjamin, Suguru Endo, William J Huggins, Ying Li, Jarrod R McClean, and Thomas E O'Brien. "Quantum error mitigation". Reviews of Modern Physics **95**, 045005 (2023).

[37] F Jauberteau and JL Jauberteau. "Numerical differentiation with noisy signal". Applied Mathematics and Computation **215**, 2283–2297 (2009).

[38] Karsten Ahnert and Markus Abel. "Numerical differentiation of experimental data: local versus global methods". Computer Physics Communications **177**, 764–774 (2007).

[39] Nicolas Boulant, Timothy F Havel, Marco A Pravia, and David G Cory. "Robust method for estimating the lindblad operators of a dissipative quantum process from measurements of the density operator at multiple time points". Physical Review A **67**, 042322 (2003).

[40] Eitan Ben Av, Yotam Shapira, Nitzan Akerman, and Roee Ozeri. "Direct reconstruction of the quantum-master-equation dynamics of a trapped-ion qubit". Physical Review A **101**, 062305 (2020).

[41] Oliver Strebel. "A preprocessing method for parameter estimation in ordinary differential equations". Chaos, Solitons & Fractals **57**, 93–104 (2013).

[42] Prithwish Bhaumik and Subhashis Ghosal. "Bayesian two-step estimation in differential equation models". Electronic Journal of Statistics **9**, 3124–3154 (2015).

[43] Yu Cao and Jianfeng Lu. "Structure-preserving numerical schemes for Lindblad equations" (2021).

[44] Gerard LG Sleijpen and Diederik R Fokkema. "Bicgstab (ell) for linear equations involving unsymmetric matrices with complex spectrum". Electronic Transactions on Numerical Analysis. **1**, 11–32 (1993).

[45] Yurii Nesterov. "Introductory lectures on convex optimization: A basic course". Volume 87. Springer Science & Business Media. (2013).

[46] Mary Beth Ruskai. "Beyond strong subadditivity? improved bounds on the contraction of generalized relative entropy". Reviews in Mathematical Physics **6**, 1147–1161 (1994).

[47] A Van der Sluis. "Stability of the solutions of linear least squares problems". Numerische Mathematik **23**, 241–254 (1974).

[48] Chi Jin, Praneeth Netrapalli, Rong Ge, Sham M Kakade, and Michael I Jordan. "A short note on concentration inequalities for random vectors with subgaussian norm" (2019).

[49] Martin Kliesch, Thomas Barthel, Christian Gogolin, Michael J Kastoryano, and Jens Eisert. "Dissipative quantum Church-Turing theorem". Physical Review Letters**107** (2011).

[50] Andrew M Childs and Tongyang Li. "Efficient simulation of sparse markovian quantum dynamics". Quantum Information & Computation **17**, 0901–0947 (2017).

[51] Richard Cleve and Chunhao Wang. "Efficient quantum algorithms for simulating Lindblad evolution". In 44th International Colloquium on Automata, Languages, and Programming, (ICALP 2017). Pages 17:1–17:14. (2017).

[52] Xiantao Li and Chunhao Wang. "Simulating Markovian Open Quantum Systems Using Higher-Order Series Expansion". In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, 50th International Colloquium on Automata, Languages, and Programming (ICALP 2023). Volume 261 of Leibniz International Proceedings in Informatics (LIPIcs), pages 87:1–87:20. Dagstuhl, Germany (2023). Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[53] András Gilyén, Srinivasan Arunachalam, and Nathan Wiebe. "Optimizing quantum optimization algorithms via faster quantum gradient computation". In Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms. Pages 1425–1444. SIAM (2019).

[54] Ke Wang and Xiantao Li (2024). url: `https://github.com/kewang-math/LearningOpenQuantumSystems`.

A Appendix

A.1 Proof of Theorem 2

Proof. The derivative of the objective function ϕ is

$$\partial_{\theta_\alpha}\phi(\boldsymbol{\theta}) = \frac{1}{N_O N_T}\sum_{k=1}^{N_O}\sum_{n=1}^{N_T}(y_{k,n}-y_{k,n}^*)\langle A^{(k)}, \partial_{\theta_\alpha}\rho(\tau_{nL};\boldsymbol{\theta})\rangle \tag{52}$$

Let $\chi_{nL}^\alpha = \partial_{\theta_\alpha}\rho(\tau_{nL};\boldsymbol{\theta})$ and $\rho_{nL} := \rho(\tau_{nL};\theta)$. The iteration formula for the density operator is of the Kraus form $\rho_{m+1} = \mathcal{K}[\rho_m] := \sum_j F_j\rho_m F_j^\dagger, m = 1,\cdots,nL$. It reveals $\chi_{m+1}^\alpha = \mathcal{K}[\chi_m^\alpha] + \partial_{\theta_\alpha}\mathcal{K}[\rho_m]$, then

$$\begin{aligned}
\langle A^{(k)}, \chi_{nL}^\alpha\rangle &= \langle A^{(k)}, \mathcal{K}\chi_{nL-1}^\alpha\rangle + \langle A^{(k)}, \partial_{\theta_\alpha}\mathcal{K}\rho_{nL-1}\rangle \\
\langle A^{(k)}, \mathcal{K}\chi_{nL-1}^\alpha\rangle &= \langle A^{(k)}, \mathcal{K}^2\chi_{nL-2}^\alpha\rangle + \langle A^{(k)}, \mathcal{K}(\partial_{\theta_\alpha}\mathcal{K})\rho_{nL-2}\rangle \\
&\vdots \\
\Rightarrow \langle A^{(k)}, \chi_{nL}^\alpha\rangle &= \langle A^{(k)}, \mathcal{K}^{nL}\chi_0^\alpha\rangle + \sum_{l=1}^{nL}\langle A^{(k)}, \mathcal{K}^{l-1}(\partial_{\theta_\alpha}\mathcal{K})\rho_{nL-l}\rangle \\
&= \sum_{l=1}^{nL}\langle A^{(k)}, \mathcal{K}^{l-1}(\partial_{\theta_\alpha}\mathcal{K})\rho_{nL-l}\rangle
\end{aligned} \tag{53}$$

as $\chi_0^\alpha = 0$. By taking the adjoint operator,

$$\begin{aligned}
\partial_{\theta_\alpha}\phi(\boldsymbol{\theta}) &= \frac{1}{N_O N_T}\sum_{k,n}(y_{k,n}-y_{k,n}^*)\sum_{l=1}^{nL}\langle(\partial_{\theta_\alpha}\mathcal{K})^*(\mathcal{K}^*)^{l-1}A^{(k)}, \rho_{nL-l}\rangle \\
&= \frac{1}{N_O N_T}\sum_{k,n}(y_{k,n}-y_{k,n}^*)\sum_{l=1}^{nL}\langle(\partial_{\theta_\alpha}\mathcal{K})^* A_{nL-l}^{(k)}, \rho_{nL-l}\rangle
\end{aligned} \tag{54}$$

where $A_{nL-1}^{(k)} := (\mathcal{K}^*)^{l-1}[A^{(k)}]$ is a back-propagated operator and the adjoint Kraus form

$$(\partial_{\theta_\alpha}\mathcal{K})^*(\rho) = \sum_j(\partial_{\theta_\alpha}F_j)^\dagger\rho F_j + F_j^\dagger\rho(\partial_{\theta_\alpha}F_j), \quad \mathcal{K}^*(\rho) = \sum_j F_j^\dagger\rho F_j \tag{55}$$

for real parameters $\boldsymbol{\theta}$. □

A.2 Second Order Implicit Approximation Algorithm

Similar to Lemma 1, we start with the unraveled SDE,

$$\mathrm{d}\,|\psi\rangle = (-iH_S\,|\psi\rangle - \frac{1}{2}\sum_{j=1}^{N_V}V_j^\dagger V_j\,|\psi\rangle)\mathrm{d}t + \sum_{j=1}^{N_V}V_j\,|\psi\rangle\,\mathrm{d}W_{j;t} = G\,|\psi\rangle\,\mathrm{d}t + \sum_{j=1}^{N_V}V_j\,|\psi\rangle\,\mathrm{d}W_{j;t} \tag{56}$$

According to the second order implicit weak scheme[25, Chapter 15], we can write down a time-marching scheme for the wave function,

$$\begin{aligned}
|\psi_{m+1}\rangle = |\psi_m\rangle &+ \frac{1}{2}(G\,|\psi_{m+1}\rangle + G\,|\psi_m\rangle)\delta t + \sum_{j=1}^{N_V}V_j\,|\psi_m\rangle\,\delta\hat{W}_j \\
&+ \frac{1}{2}\sum_{j=1}^{N_V}L^0V_j\,|\psi_m\rangle\,\delta\hat{W}_j\delta t + \frac{1}{2}\sum_{j_1,j_2=1}^{N_V}L^{j_1}V_{j_2}\,|\psi_m\rangle\,(\delta\hat{W}_{j_1}\delta\hat{W}_{j_2} + U_{j_1,j_2}).
\end{aligned} \tag{57}$$

In this expression, $\delta\hat{W}$ is an approximation of an increment of the Brownian motion. It was suggested in [25] to sample with the three-point distribution,

$$P(\delta\hat{W} = \pm\sqrt{3\Delta}) = \frac{1}{6}, P(\delta\hat{W} = 0) = \frac{2}{3} \tag{58}$$

Meanwhile, U_{j_1,j_2}'s are independent two-point distributed random variables with

$$\begin{aligned} &P(U_{j_1,j_2} = \pm\Delta) = \frac{1}{2}, \forall j_2 = 1, \cdots, j_1 - 1 \\ &U_{j_1,j_1} = -\Delta \\ &U_{j_1,j_2} = -U_{j_2,j_1}, \text{ for } j_2 = j_1 + 1, \cdots, N_V, j_1 = 1, \cdots, N_V. \end{aligned} \tag{59}$$

In Eq. (57), the diffusion operator L^{j_1}'s are defined as follows.

$$\begin{aligned} L^0 V_j |\psi\rangle &= \sum_{k=1}^{d} (G\psi)^k \frac{\partial}{\partial \psi^k} V_j |\psi\rangle = \sum_{k=1}^{d} (G\psi)^k V_j(:,k) = V_j G |\psi\rangle \\ L^{j_1} V_{j_2} |\psi\rangle &= \sum_{k=1}^{d} (V_{j_1}\psi)^k \frac{\partial}{\partial \psi^k} V_{j_2} |\psi\rangle = \sum_{k=1}^{d} (V_{j_1}\psi)^k V_{j_2}(:,k) = V_{j_2} V_{j_1} |\psi\rangle \end{aligned} \tag{60}$$

Here the notation $(:,k)$ indicates the kth column of the matrix. Thus, Eq. (20) or Eq. (23) can be rewritten as

$$\begin{aligned} |\psi_{m+1}\rangle = &(I - \frac{1}{2}G\delta t)^{-1}(I + \frac{1}{2}G\delta t) |\psi_m\rangle + \sum_{j=1}^{N_V} (I - \frac{1}{2}G\delta t)^{-1}(V_j + \frac{1}{2}V_j G\delta t) |\psi_m\rangle \delta\hat{W}_j \\ &+ \frac{1}{2}(I - \frac{1}{2}G\delta t)^{-1} \sum_{j_1,j_2=1}^{N_V} V_{j_2} V_{j_1} |\psi_m\rangle (\delta\hat{W}_{j_1}\delta\hat{W}_{j_2} + U_{j_1,j_2}). \end{aligned} \tag{61}$$

By taking expectations, this time-marching scheme induces an iteration formula for the density operator $\rho_m = \mathbb{E}[|\psi_m\rangle \langle\psi_m|]$, as follows.

$$\begin{aligned} \rho_{m+1} = &(I - \frac{1}{2}G\delta t)^{-1}(I + \frac{1}{2}G\delta t)\rho_m(I + \frac{1}{2}G\delta t)(I - \frac{1}{2}G\delta t)^{-1} \\ &+ (I - \frac{1}{2}G\delta t)^{-1} \sum_{j=1}^{N_V} V_j(I + \frac{1}{2}G\delta t)\rho_m(I + \frac{1}{2}G\delta t)V_j^\dagger(I - \frac{1}{2}G\delta t)^{-1}\delta t \\ &+ \frac{1}{2}(I - \frac{1}{2}G\delta t)^{-1} \sum_{j_1,j_2=1}^{N_V} V_{j_1} V_{j_2}\rho_m V_{j_2\dagger} V_{j_1\dagger}(I - \frac{1}{2}G\delta t)^{-1}(\delta t)^2 \\ = &\sum_{j=0}^{N_V} F_j \rho_m F_j^\dagger + \sum_{j_1,j_2=1}^{N_V} A_{j_1,j_2}\rho_m A_{j_1,j_2}^\dagger \\ = &\sum_{j=0}^{N_V^2+N_V} F_j \rho_m F_j^\dagger = \mathcal{K}[\rho_m] \end{aligned} \tag{62}$$

where the Kraus operator

$$\begin{aligned} F_0 &= (I - \frac{1}{2}G\delta t)^{-1}(I + \frac{1}{2}G\delta t) \\ F_j &= (I - \frac{1}{2}G\delta t)^{-1} V_j (I + \frac{1}{2}G\delta t)\sqrt{\delta t}, j = 1, \cdots, N_V \\ A_{j_1,j_2} &= \frac{1}{\sqrt{2}}(I - \frac{1}{2}G\delta t)^{-1} V_{j_1} V_{j_2}\delta t, j_1, j_2 = 1, \cdots, N_V \\ F_{j_1+N_V j_2} &= A_{j_1,j_2}, j_1, j_2 = 1, \cdots, N_V \end{aligned} \tag{63}$$

To show that the method has second order accuracy, we first expand the solution of the Lindblad equation (3),

$$\rho(\tau_{m+1}) = \rho(\tau_m) + \delta t\mathcal{L}\rho(\tau_m) + \frac{1}{2}\delta t^2\mathcal{L}^2\rho(\tau_m) + \mathcal{O}(\delta t^3). \tag{64}$$

In particular, we have,

$$\mathcal{L}\rho = -i[H,\rho] + \sum_{j=1}^{N_V}\left(V_j\rho V_j^\dagger - \frac{1}{2}\{V_j^\dagger V_j,\rho\}\right).$$

We now show that the method (20) has a consistent expansion at (64) up to $\mathcal{O}(\delta t^2)$. Toward this end, we expand the Kraus operators,

$$\begin{aligned}
F_0 &= I + G\delta t + \frac{1}{2}G^2\delta t^2 + \mathcal{O}(\delta t^3),\\
F_j &= V_j\sqrt{\delta t} + \frac{1}{2}\{G,V_j\}\delta t\sqrt{\delta t} + \frac{1}{2}\{G,GV_j\}\delta t^2\sqrt{\delta t} + \mathcal{O}(\delta t^{7/2}), \qquad j = 1,\cdots,N_V\\
F_{j_1+N_V j_2} &= \frac{1}{\sqrt{2}}V_{j_1}V_{j_2}\delta t + \mathcal{O}(\delta t^{3/2}), \qquad j_1,j_2 = 1,\cdots,N_V
\end{aligned}$$

By substituting these expansions of the jump operators, and only keeping terms of order up to $\mathcal{O}(\delta t^2)$, we find that,

$$\begin{aligned}
\rho_{m+1} =& \rho_m + \delta t(G\rho_m + \rho_m G^\dagger) + \frac{1}{2}\delta t^2(G^2\rho_m + 2G\rho_m G^\dagger + \rho_m {G^\dagger}^2)\\
&+ \delta t\sum_{j=1}^{N_V} V_j\rho_m V_j^\dagger + \frac{1}{2}\delta t^2\sum_{j=1}^{N_V}\left(V_j\rho_m\{G,V_j\}^\dagger + \{G,V_j\}\rho_m V_j^\dagger\right)\\
&+ \frac{1}{2}\delta t^2\sum_{j_1,j_2=1}^{N_V} V_{j_1}V_{j_2}\rho_m V_{j_2}^\dagger V_{j_1}^\dagger + \mathcal{O}(\delta t^3).
\end{aligned}$$

In light of Eq. (14), the $\mathcal{O}(\delta t)$ terms are consistent with that in (64). With lengthy calculations, we can also see that the $\mathcal{O}(\delta t^2)$ terms are also consistent. From the calculation of the consistency of $\mathcal{O}(\delta t)$, we know that $G\rho_m + \rho_m G^\dagger = \mathcal{L}\rho_m - \sum_j V_j\rho_m V_j^\dagger$. The detailed calculation can be summarized as follows:

$$\begin{aligned}
&G^2\rho_m + 2G\rho_m G^\dagger + \rho_m {G^\dagger}^2 = G(G\rho_m + \rho_m G^\dagger) + (G\rho_m + \rho_m G^\dagger)G^\dagger\\
=&\mathcal{L}^2\rho_m - \sum_{j=1}^{N_V}\mathcal{L}(V_j\rho_m V_j^\dagger) - \sum_{j=1}^{N_V} V_j\mathcal{L}\rho_m V_j^\dagger + \sum_{j_1,j_2=1}^{N_V} V_{j_2}V_{j_1}\rho_m V_{j_1}V_{j_2}\\
&\sum_{j=1}^{N_V}\left(V_j\rho_m\{G,V_j\}^\dagger + \{G,V_j\}\rho_m V_j^\dagger\right) = \sum_j(V_j\rho_m V_j^\dagger G^\dagger + GV_j\rho_m V_j^\dagger) + V_j(\rho_m G^\dagger + G\rho_m)V_j^\dagger\\
=&\sum_j\mathcal{L}(V_j\rho_m V_j^\dagger) - 2\sum_{j_1,j_2} V_{j_2}V_{j_1}\rho_m V_{j_1}^\dagger V_{j_2}^\dagger + \sum_j V_j\mathcal{L}\rho_m V_j^\dagger\\
\Rightarrow& G^2\rho_m + 2G\rho_m G^\dagger + \rho_m {G^\dagger}^2 + \sum_{j=1}^{N_V}\left(V_j\rho_m\{G,V_j\}^\dagger + \{G,V_j\}\rho_m V_j^\dagger\right) + \sum_{j_1,j_2} V_{j_2}V_{j_1}\rho_m V_{j_1}^\dagger V_{j_2}^\dagger = \mathcal{L}^2\rho_m.
\end{aligned}$$

This completes the proof.