Quantum algorithm to simulate Lindblad master equations

Evan Borras [1,2] and Milad Marvian [1,2,3]
[1] *Center for Quantum Information and Control, University of New Mexico, Albuquerque, New Mexico 87131, USA*
[2] *Department of Physics and Astronomy, University of New Mexico, Albuquerque, New Mexico 87131, USA*
[3] *Department of Electrical & Computer Engineering, University of New Mexico, Albuquerque, New Mexico 87131, USA*

(Received 2 July 2024; accepted 4 April 2025; published 22 April 2025)

We present a quantum algorithm for simulating a family of Markovian master equations that can be realized through a probabilistic application of unitary channels and state preparation. Our approach employs a second-order product formula for the Lindblad master equation, achieved by decomposing the dynamics into dissipative and Hamiltonian components and replacing the dissipative segments with randomly compiled, easily implementable elements. The sampling approach eliminates the need for ancillary qubits to simulate the dissipation process and reduces the gate complexity in terms of the number of jump operators. We provide a rigorous performance analysis of the algorithm. We also extend the algorithm to time-dependent Lindblad equations, generalize the family of Markovian master equations it can be applied to, and explore applications beyond the Markovian noise model. A new error bound, in terms of the diamond norm, for second-order product formulas for time-dependent Liouvillians is provided that might be of independent interest.

DOI: 10.1103/PhysRevResearch.7.023076

I. INTRODUCTION

One of the first suggested applications of quantum computing was to simulate quantum mechanical systems [1]. Since Feynman's initial proposal, the field of quantum simulation has grown rapidly, resulting in numerous algorithms that offer asymptotic speedups over the best possible classical algorithms. Initially, much focus has been on simulating closed quantum systems [2–9], only recently has there been a significant interest in developing quantum algorithms for simulating open quantum systems [10–23]. Notable examples are algorithms to simulate a single quantum channel applied to an input quantum state [11,15,19], algorithms that simulate continuous-time dynamics describable by master equations [10–14,16,20]. Special attention is paid to dynamics generated by time-independent Lindblad master equations, since they cover the common scenario of a quantum system weakly coupled to a Markovian environment. Some examples include Refs. [10–14,16,20–22] with [13] achieving the current state-of-the-art asymptotic scaling. Similar to Hamiltonian simulation, the current classical Lindblad simulation algorithms are inefficient, thus efficient quantum algorithms for Lindblad simulation could prove useful for future scientific needs.

Interestingly, many of the current Lindblad simulation algorithms share common features. They deterministically implement the full set of jump operators for the Lindbladian and treat both the dissipative and unitary dynamics on equal footing. In addition, the resources required by these algorithms explicitly increase (polynomially) with respect to the number of Lindblad jump operators along with the overall norm of the Lindbladian. A missing feature in the existing algorithms is taking advantage of the fact that in many realistic open system dynamics, there are classical uncertainties regarding the effect of the environment on the system. This noise-agnostic approach can lead to situations where the noisy open system simulation algorithms can scale worse, sometimes even exponentially worse, than simulating the noiseless system. For example, naive algorithms to simulate Hamiltonian evolution in the presence of global depolarizing noise would require exponential time for many of the general-purpose open quantum system algorithms, given the fact that global depolarization requires exponentially many jump operators. However, such an explicit scaling with the number of jump operators can be reduced if the classical randomness, present in the description of the open system dynamics, is directly incorporated into the design of the algorithm to simulate dissipation. This strategy has recently been considered for a restricted class of Lindbladians, in which a first-order approximation to a dissipator is used for thermal state preparation on a quantum computer [24]. It has also been used to simulate Lindblad dynamics through leveraging the intrinsic noise of quantum processors [21,22], as well as in the design of an algorithm that uses a classical processor alongside a quantum computer to reconstruct the dynamics based on a series of sampled states [25].

In this paper, we present a quantum algorithm for simulating both the coherent and dissipative evolution of a Lindbladian, with no need for any ancilla qubits and reduced gate complexity in the number of jump operators, although for a restricted class of Lindbladians. The class of Lindbladians we consider are sums of a Hamiltonian part and a dissipator, where the dissipator is constrained to generate channels that are convex combinations of unitary channels and fixed state preparation. Although our noise model is restrictive, it does encapsulate many physically relevant dissipative processes such as both local and global depolarizing noise, dephasing and bit-flip noise, and other random Pauli channels. The algorithm we introduce uses a second-order product formula to divide the generator into its Hamiltonian part and dissipator. We then simulate the dissipative portion using a sampling protocol, with the Hamiltonian dynamics being simulated using any Hamiltonian simulation algorithm preferred. The only ancillary qubits needed by the algorithm are what is needed to implement the desired Hamiltonian simulation subroutine. The scaling of the circuit with respect to simulation time and precision is $O(T^{1.5}/\sqrt{\epsilon})$, where T is the total simulation time and ϵ is the accuracy in the diamond norm. This is an improvement over the $O(T^2/\epsilon)$ scaling of the randomized Lindblad simulation algorithm presented in Ref. [24], which also eliminates the circuits scaling on the number of jump operators.

We also generalize the algorithm to simulate a family of time-dependent Lindbladians. To analyze the performance in the time-dependent case, we prove a new second-order product formula for time-dependent Louvillians, consisting of an arbitrary time-dependent and an arbitrary time-independent term, which might be of independent interest. Additionally, we explore how limited ancillary qubits can broaden the scope of master equations that our algorithm can simulate, for example simulating local amplitude damping noise using a few ancilla qubits. We also discuss methods to extend the algorithm to non-Markovian master equations. Finally, we conclude with providing a few demonstrations of our algorithm applied to various master equations.

II. STOCHASTICALLY SIMULATABLE CHANNELS

Let $\mathcal{E}$ be a CPTP map that can be simulated stochastically, i.e., it can be implemented by sampling a set of quantum operations $\{\mathcal{A}_i\}$ according to probabilities p_i such that $\mathcal{E}(\rho) = \mathbb{E}[\mathcal{A}_i(\rho)]$ holds. Then the measurement statistics of any quantum circuit composed of multiple applications of $\mathcal{E}$ can be reproduced by replacing any application of $\mathcal{E}$ with an implementation of $\mathcal{A}_i$ sampled independently. In this work we are interested in estimating the expected value of a general observable using our simulation algorithm. With the goal of designing a low-cost quantum algorithm, we restrict the operations $\mathcal{A}_i$ to be either unitary or fixed state preparation operations. More specifically, we consider stochastically simulatable channels as defined below.

Definition 1. We call $\mathcal{E}$ a stochastically simulatable channel if it can be written as:

$$\mathcal{E}(\rho) = q\left(\sum_i \lambda_i U_i \rho U_i^\dagger\right) + (1-q)\rho_f, \tag{1}$$

where U_i are unitary operators and ρ_f is a fixed quantum state. Scalars q and λ_i are constrained such that $0 \leqslant q \leqslant 1$, $0 \leqslant \lambda_i \leqslant 1$, and $\sum_i \lambda_i = 1$.

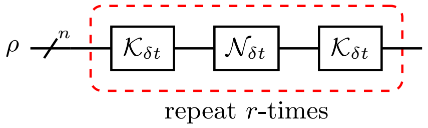

FIG. 1. Simulation gadget.

The cost of implementing $\mathcal{E}$ solely depends on the complexity of implementing $\{U_i\}$ and also the preparation of ρ_f. Ideally these operations should be efficiently implementable. Some examples include when $\{U_i\}$ are n-fold tensor products of single-qubit unitaries and also when they are generated by Clifford circuits.

In this work we mainly focus on simulating open quantum systems described by a Lindblad master equation, given as $\frac{d\rho}{dt} = \mathcal{L}(\rho)$ with $\mathcal{L} = \mathcal{H} + \mathcal{D}$, where $\mathcal{H}(\rho) = -i[H, \rho]$ generates Hamiltonian evolution and the dissipator $\mathcal{D}(\rho) = \sum_{\mu=1}^m (L_\mu \rho L_\mu^\dagger - \frac{1}{2}\{L_\mu^\dagger L_\mu, \rho\})$ models the system-environment interactions. Here H is the system Hamiltonian and L_μ are the Lindblad's jump operators.

An example of a class of Lindblad superoperators that satisfy Definition 1 for a small interval of time are dissipators $\mathcal{D}$ with jump operators $L_\mu = \alpha_\mu U_\mu$ where U_μ is unitary. It is straightforward to check that in this case, when $\delta t \|\mathcal{D}\|_\diamond \leqslant 1$, the operator $e^{\delta t \mathcal{D}}$ can be approximated by a random unitary channel with arbitrarily small error, see Appendix A. When U_μ belongs to a group of efficiently implementable unitaries, such as n-fold tensor products of single-qubit unitaries or Clifford circuits, then the random unitary channel fits our ideal scenario discussed above, since the complexity of implementation does not change when the unitaries are composed. This class of Lindblad master equations captures many useful families including random Pauli channels.

III. NEW ALGORITHM

We now introduce the quantum algorithm to simulate Lindbladian dynamics. In its simplest form, given a time-independent Lindblad superoperator $\mathcal{L} = \mathcal{H} + \mathcal{D}$ where $\mathcal{H}$ is the generator of the Hamiltonian dynamics and $\mathcal{D}$ is the generator of a (approximately) stochastically simulatable map, our proposed algorithm simulates the evolution of an arbitrary input state ρ under $\mathcal{L}$ up to time T and precision ϵ by repeatedly implementing Lindbladian simulation gadgets depicted in Fig. 1.

Each Lindbladian simulation gadget is built of two different subroutines, one implementing the map $\mathcal{K}_{\delta t}$, which approximates $e^{\frac{\delta t}{2}\mathcal{H}}$ and can be executed using any choice of Hamiltonian simulation algorithm, and the other, $\mathcal{N}_{\delta t}$ which approximates $e^{\delta t \mathcal{D}}$ in expectation using a simple sampling routine.

As discussed, we assume $\mathcal{D}$ generates an approximately stochastically simulatable map over a small period of time δt. Thus we can we write $e^{\delta t \mathcal{D}} = \mathcal{N}_{\delta t} + \mathcal{O}(\delta t^3)$, where $\mathcal{N}_{\delta t}$ is a stochastically simulatable map, i.e., satisfying Definition 1. Since $\mathcal{N}_{\delta t}$ satisfies Definition 1, we can write

$$\mathcal{N}_{\delta t}(\rho) = q(\delta t) \sum_i \lambda_i(\delta t) U_{\delta t}^{(i)} \rho U_{\delta t}^{(i)\dagger} + [1 - q(\delta t)]\rho_{\delta t}. \tag{2}$$

Defining quantum operations

$$\mathcal{A}^{(i)}_{\delta t}(\rho) = \begin{cases} P_{\delta t}(\rho) = \rho_{\delta t} & \text{for } i = 0, \\ U^{(i)}_{\delta t}\rho U^{(i)\dagger}_{\delta t} & \text{otherwise ,} \end{cases} \tag{3}$$

and probabilities

$$p_i(\delta t) = \begin{cases} 1 - q(\delta t) & \text{for } i = 0, \\ q(\delta t)\lambda_i(\delta t) & \text{otherwise.} \end{cases} \tag{4}$$

we can rewrite Eq. (2) as

$$\mathcal{N}_{\delta t}(\rho) = \sum_i p_i(\delta t)\mathcal{A}^{(i)}_{\delta t}(\rho) = \mathbb{E}_{i\sim p_i(\delta t)}\big[\mathcal{A}^{(i)}_{\delta t}(\rho)\big]. \tag{5}$$

The above equation suggests a simple method to implement our approximation to the map $e^{\delta t\mathcal{D}}$, simply sample $i \sim p_i(\delta t)$ and implement the operation $\mathcal{A}^{(i)}_{\delta t}$. Thus the gate complexity of $\mathcal{N}_{\delta t}$ is independent of the number of jump operators and only depends on the complexity of implementing unitaries $U^{(i)}_{\delta t}$ and fixed state preparation oracle $P_{\delta t}$, which replaces the state of the system with a fixed state. This procedure assures the measurement statistics over many different compilations of the quantum circuit implementing $\mathcal{N}_{\delta t}$ will replicate the measurement statistics of a single circuit implementing $\mathcal{N}_{\delta t}$ exactly.

Thus when compiling each of the Lindbladian simulation gadgets, we independently sample $i \sim p_i(\delta t)$ and implement the operation $\mathcal{A}^{(i)}_{\delta t}$ in place of $\mathcal{N}_{\delta t}$ in the gadget. The full algorithm consists of sequentially repeating the blocks of $\mathcal{K}_{\delta t}$ using any closed-system quantum simulation algorithm of choice, and sampling $\mathcal{A}^{(i)}_{\delta t}$ for each $\mathcal{N}_{\delta t}$ block. The number of repetitions r, required to guarantee a desired accuracy ϵ, would determine the query complexity of the algorithm.

A. Algorithm analysis

At a high level, to analyze the performance of the algorithm, we leverage a product formula for Lindblad superoperators to split up the dynamics into its Hamiltonian part and a stochastically simulatable dissipator. The total simulation time and precision then constrain the individual subroutines' precision and simulation time. Since we can implement $\mathcal{N}_{\delta t}$ without error, our constraints only apply to the Hamiltonian simulation subroutine $\mathcal{K}_{\delta t}$.

First, we introduce some notation. The diamond norm of a linear map $\mathcal{L}$ over linear operators on a finite-dimensional Hilbert space is defined as $\|\mathcal{L}\|_\diamond = \max_{\|\rho\|_1=1}\|\mathcal{I}\otimes\mathcal{L}(\rho)\|_1$, where $\|\cdot\|_1$ denotes the trace norm and $\mathcal{I}$ the identity map. Following Ref. [20] we define $\|\mathcal{L}\|_{\text{pauli}}$ of a Lindblad superoperator $\mathcal{L}$ as $\|\mathcal{L}\|_{\text{pauli}} = \sum_{k=0}^{s-1}\beta_{0k} + \sum_{\mu=1}^{m}(\sum_{k=0}^{s-1}\beta_{\mu k})^2$, where $H = \sum_{k=0}^{s-1}\beta_{0k}V_{0k}$, and $L_\mu = \sum_{k=0}^{s-1}\beta_{\mu k}V_{\mu k}$ with $\beta_{\mu k} > 0$ and $V_{\mu k}$ an n-fold tensor product of Pauli operators with arbitrary phases. Finally, we denote the total simulation time and precision by T and ϵ, respectively.

We proceed by finding an upper bound on the error between r applications of our subroutines and the true evolution generated by $\mathcal{L}$. Assuming $(\frac{\|\mathcal{H}\|_\diamond}{2} + \|\mathcal{D}\|_\diamond)\delta t \leqslant 1$, we can apply a second-order product formula [26] (see Theorem 2 for our generalized version), alongside the submultiplicative and additive properties of the diamond norm [27] to get:

$$\|e^{T\mathcal{L}} - (\mathcal{K}_{\delta t}\circ\mathcal{N}_{\delta t}\circ\mathcal{K}_{\delta t})^r\|_\diamond \leqslant \frac{\|[\mathcal{H},\mathcal{D}]\|_\diamond}{3}\left(\frac{\|\mathcal{H}\|_\diamond}{2} + \|\mathcal{D}\|_\diamond\right)r\delta t^3 + 2r(\epsilon_H + \epsilon_D), \tag{6}$$

where $\epsilon_H = \|e^{\frac{\delta t}{2}\mathcal{H}} - \mathcal{K}_{\delta t}\|_\diamond$ and $\epsilon_D = \|e^{\delta t\mathcal{D}} - \mathcal{N}_{\delta t}\|_\diamond$, which represent the errors each of our subroutines incur.

To formalize our assumption about the generator $\mathcal{D}$, we require

$$\epsilon_D = \|e^{\delta t\mathcal{D}} - \mathcal{N}_{\delta t}\|_\diamond \leqslant c_0(\|\mathcal{D}\|_\diamond\delta t)^3, \tag{7}$$

where c_0 is a constant determined by the specifics of the dissipator $\mathcal{D}$. As discussed before, dissipators with jump operators of the form $L_\mu = \alpha_\mu U_\mu$ can be approximated by a stochastically simulatable channel up to arbitrary small error. Such dissipators are an example of when requirement (7) is satisfied. Using $\|\mathcal{L}\|_\diamond \leqslant 2\|\mathcal{L}\|_{\text{pauli}}$ from [20] to carry out the upper bounds in terms of $\|\cdot\|_{\text{pauli}}$ gives us a way to easily compare to the state-of-the-art Lindblad simulation algorithms [14,20,25]. Thus our bound becomes,

$$\|e^{T\mathcal{L}} - (\mathcal{K}_{\delta t}\circ\mathcal{N}_{\delta t}\circ\mathcal{K}_{\delta t})^r\|_\diamond \leqslant \tfrac{8}{3}r(1+6c_0)(\|\mathcal{L}\|_{\text{pauli}}\delta t)^3 + 2r\epsilon_H. \tag{8}$$

To keep the precision of the total simulation over time T within total error of $O(\epsilon)$ Eq. (8) implies we must take $r = O(\sqrt{\frac{1+6c_0}{\epsilon}}(\|\mathcal{L}\|_{\text{pauli}}T)^{1.5})$, and $\epsilon_H = O(\epsilon/r)$. All that is left is to design an implementation of the Hamiltonian simulation subroutine.

Choosing to use the truncated Taylor series algorithm of Ref. [7] provides the gate complexity for the subroutine to be $\tilde{O}(\delta t\|\mathcal{H}\|_{\text{pauli}}sn\log(\frac{\delta t\|\mathcal{H}\|_{\text{pauli}}}{\epsilon_H}))$, where n is the number of qubits and s is the number of Pauli terms in the Hamiltonian. The total gate complexity is just the subroutine's cost multiplied by r. Since subroutine $\mathcal{N}_{\delta t}$ does not require any ancilla to implement, the only ancilla cost of the algorithm comes from our implementation of $\mathcal{K}_{\delta t}$. Assuming we can reset the ancilla registers before each $\mathcal{K}_{\delta t}$ subroutine, algorithm of Ref. [7] gives us the ancilla cost for implementing $\mathcal{K}_{\delta t}$ to be $\tilde{O}[\log(s)\log(\delta t/\epsilon_H)]$.

Alternatively, we could use an algorithm based on product formulas [28,29] to implement the Hamiltonian simulation subroutine $\mathcal{K}_{\delta t}$, which would remove the need for any ancilla, at the expense of a worse gate complexity to simulate the Hamiltonian dynamics. We can summarize the analysis in the following theorem.

Theorem 1. Let $\mathcal{L} = \mathcal{H} + \mathcal{D}$ be a Lindbladian with Hamiltonian portion $\mathcal{H}$ and dissipative portion $\mathcal{D}$. Assume $\mathcal{D}$ generates an approximately stochastically simulatable channel satisfying (7). Then there exists a quantum algorithm that simulates $e^{T\mathcal{L}}$ up to precision ϵ that requires

$$r = O\left(\sqrt{\frac{1+6c_0}{\epsilon}}(\|\mathcal{L}\|_{\text{pauli}}T)^{1.5}\right) \tag{9}$$

calls to oracles $\mathcal{A}^{(i)}_{\delta t}$, and any Hamiltonian simulation subroutine $\mathcal{K}_{\delta t}$ where $\delta t = T/r$. No ancilla qubits are needed beyond what is required for the Hamiltonian simulation subroutine $\mathcal{K}_{\delta t}$.

IV. EXTENSIONS

The algorithm described above can only simulated Lindbladians, which are time independent and satisfy our constraint of being composed of a dissipator, which generates a channel that approximately satisfies Definition 1. In the sections to follow we show how the algorithm described above can be generalized to include Lindbladians with dissipators that are time dependent, how to go beyond the constraint of Definition 1, as well as how to incorporate certain kinds of non-Markovian dissipation.

A. Time-dependent Lindblad equation

The algorithm and analysis above can be generalized to include time-dependent Lindblad superoperators with a time-dependent dissipative portion $\mathcal{D}(t)$. If the dissipator generates a channel that is approximately stochastically simulatable at all times, i.e.,

$$\mathcal{T}e^{\int_t^{t+\delta t}\mathcal{D}(t')dt'}[\rho(t)] = \mathcal{N}_{t,\delta t}[\rho(t)] + \mathcal{O}(\delta t^3), \tag{10}$$

holds for all t, then the algorithm can be easily modified to simulate such systems. To do so we leverage Theorem 2 to divide the time-dependent dynamics into its time-dependent dissipator and its Hamiltonian part.

Theorem 2. Let $\mathcal{H}$ and $\mathcal{D}(t)$ be Lindbladians and $0 \leqslant (\frac{\|\mathcal{H}\|_\diamond}{2} + \sup_t \|\mathcal{D}(t)\|_\diamond)\delta t \leqslant 1$, then

$$\begin{aligned}&\left\|\mathcal{T}e^{\int_t^{t+\delta t}(\mathcal{H}+\mathcal{D}(t'))dt'} - e^{\frac{\delta t}{2}\mathcal{H}} \circ \mathcal{T}e^{\int_t^{t+\delta t}\mathcal{D}(t')dt'} \circ e^{\frac{\delta t}{2}\mathcal{H}}\right\|_\diamond \\ &\quad \leqslant \frac{\sup_t \|[\mathcal{H},\mathcal{D}(t)]\|_\diamond}{3}\left(\frac{\|\mathcal{H}\|_\diamond}{2}\sup_t \|\mathcal{D}(t)\|_\diamond\right)\delta t^3.\end{aligned} \tag{11}$$

Proof. See Appendix B. ■

Similar to the time-independent case, we implement subroutines to approximate the dynamics generated by these two generators. The only difference lies in how we implement the dissipative subroutine. In the time-dependent case, we implement the stochastically simulatable channels $\mathcal{N}_{t,\delta t}$ as our subroutine approximating the dissipative dynamics. Since $\mathcal{N}_{t,\delta t}$ explicitly depends on time, we will implement different stochastically simulatable channels at each time step t where we require a dissipative subroutine. The rest of the analysis of the algorithm is similar to the time-independent case. The total costs as shown above hold in the time-dependent scenario with the only difference being, replacing $\|\mathcal{L}\|_{\text{pauli}}$ with $\sup_t \|\mathcal{L}(t)\|_{\text{pauli}}$ and c_0 with $\sup_t c_0(t)$ in Theorem 1.

B. Generalized noise model via ancilla qubits

At the expense of adding ancilla qubits and applying unitary operations on the joint system, one can extend the scope of the proposed method to simulate a larger family of noise models. The Stinespring dilation theorem guarantees that any CPTP map acting on $2^l \times 2^l$ density operators can be implemented by adding $2l$ ancilla qubits and performing a (generally complicated) unitary operation on the joint system [30]. However, the probabilistic implementation of channels can reduce this ancilla requirement.

Remark 1. The algorithm can implement any CPTP map acting on $2^l \times 2^l$ density operators using at most l ancilla qubits.

This observation follows from the fact that any such a map can be expressed as a convex combination of extremal channels with each extremal channel only having at most 2^l many Kraus operators [31]. Therefore implementation of dilatation of any of the extremal channels requires at most l ancilla qubits. Given that only one extremal channel needs to be implemented in each compilation, at most l ancilla suffices in each run. The ancilla qubits can be reset after the application of the channel.

Simulation of local noise, naturally arising in many physical systems, is of particular interest. A local (but possibly correlated) noise model can be described by a CPTP map,

$$\mathcal{E}(\rho) = \sum_i \lambda_i \mathcal{G}_0^{(i)} \otimes \ldots \otimes \mathcal{G}_{n-1}^{(i)}(\rho), \tag{12}$$

where each $\mathcal{G}_j^{(i)}$ acts on the jth qubit, with $l = 1$. If each $\mathcal{G}_j^{(i)}$ is unital then we can expand each operation in terms of a convex combination of single-qubit unitaries. This gives us a description of the operation that satisfies Definition 1 where the $U_i(t)$ operations are n-fold tensor products of single-qubit unitaries, thus efficiently implementable. If $\mathcal{G}_j^{(i)}$ are not unital, for example such as amplitude damping, then we only need one ancilla rather than two, to implement each of the local maps. Therefore the full noisy process $\mathcal{E}(\rho)$ can be simulated by adding at most n extra ancilla qubits and two-qubit gates, which is less than the $2n$ ancilla needed if we chose to directly dilate.

C. Incorporating time correlations

As discussed, to simulate a time-dependent Markovian master equation, we implement each dissipation block of $\mathcal{N}_{t,\delta t}$ by generating samples independent from the other blocks. However, it is straightforward to modify the algorithm to simulate dynamics that have correlations in time by simply introducing correlations in the sampling of dissipation blocks used in the compilation. In particular, the probabilities and operations in Eqs. (3) and (4) can be generalized to define joint probabilities and operations over multiple time steps. Therefore, the domain of master equations simulable by the presented algorithm goes beyond the Markovian master equation we focused on in this work.

V. DEMONSTRATIONS

Given the algorithm and extensions described above, we provide three different demonstrations of our algorithm to follow. For the first example we consider an arbitrary Hamiltonian simulation problem on n qubits under both global and local depolarizing noise. Although global depolarization is not the most physically relevant noise model for systems, it suffices to demonstrate that our algorithm can handle a dissipator with an exponential number of jump operators. The second example we provide is the simulation of crosstalk on superconducting transmon qubits. Finally we describe how $1/f$ noise on superconducting transmon qubits can be simulated using the extension of our algorithm to dissipation that correlates in time.

A. Hamiltonian simulation under depolarizing dissipation

Consider the case of a Lindbladian $\mathcal{L}$, which generates arbitrary Hamiltonian evolution along with global depolarizing noise,

$$\begin{aligned}\mathcal{L}(\rho) &= \mathcal{H}(\rho) + \mathcal{D}(\rho)\\ &= -i[H,\rho] + \sum_{\mu=1}^{m}\left(L_\mu \rho L_\mu^\dagger - \frac{1}{2}\{L_\mu^\dagger L_\mu, \rho\}\right)\\ &= -i[H,\rho] + \sum_{\mu=1}^{4^n-1}\frac{\gamma}{4^n}(V_\mu\rho V_\mu^\dagger - \rho), \end{aligned} \tag{13}$$

where H is an arbitrary Hamiltonian, $\gamma \in \mathbb{R}_+ \setminus \{0\}$, and $V_\mu \in \{\mathbb{I}, X, Y, Z\}^{\otimes n} \setminus \{\mathbb{I}^{\otimes n}\}$. There are $m = 4^n - 1$ jump operators for $\mathcal{D}_\gamma$ of the form $L_\mu = \sqrt{\frac{\gamma}{4^n}}V_\mu$. Since V_μ is unitary, we know from Appendix A that $e^{\delta t\mathcal{D}}$ is approximately stochastically simulatable up to arbitrary precision. Thus oracles $\mathcal{A}_{\delta t}^{(i)}$ are compositions of elements in $\{\mathbb{I}, X, Y, Z\}^{\otimes n} \setminus \{\mathbb{I}^{\otimes n}\}$ implying $\mathcal{A}_{\delta t}^{(i)} \in \{\mathbb{I}, X, Y, Z\}^{\otimes n}$. Calculating $\|\mathcal{L}\|_{\text{pauli}}$ gives us

$$\begin{aligned}\|\mathcal{L}\|_{\text{pauli}} &= \|\mathcal{H}\|_{\text{pauli}} + \|\mathcal{D}_\gamma\|_{\text{pauli}}\\ &= \|\mathcal{H}\|_{\text{pauli}} + \sum_{\mu=1}^{m}\left(\sum_{k=0}^{s-1}\beta_{\mu k}\right)^2. \end{aligned} \tag{14}$$

Since the jump operators $L_\mu = \sqrt{\frac{\gamma}{4^n}}V_\mu$ are already in the form of $L_\mu = \sum_{k=0}^{s-1}\beta_{\mu k}V_{\mu k}$ where $\beta_{\mu k} > 0$ and $V_{\mu k}$ is an n-fold tensor product of Pauli operators with arbitrary phase, we find

$$\sum_{k=0}^{s-1}\beta_{\mu k} = \sqrt{\frac{\gamma}{4^n}}. \tag{15}$$

This implies

$$\begin{aligned}\|\mathcal{L}\|_{\text{pauli}} &= \|\mathcal{H}\|_{\text{pauli}} + \sum_{\mu=1}^{4^n-1}\frac{\gamma}{4^n}\\ &= \|\mathcal{H}\|_{\text{pauli}} + \left(1 - \frac{1}{4^n}\right)\gamma\\ &= O(\|\mathcal{H}\|_{\text{pauli}} + \gamma). \end{aligned} \tag{16}$$

Since $\mathcal{A}_{\delta t}^{(i)} \in \{\mathbb{I}, X, Y, Z\}^{\otimes n}$, each $\mathcal{A}_{\delta t}^{(i)}$ is an n-fold tensor products of single-qubit Pauli operators. Thus each $\mathcal{A}_{\delta t}^{(i)}$ requires $O(n)$ single-qubit gates to implement. Applying Theorem 1 gives us the total cost to simulate the dynamics described by $\mathcal{L}_\gamma$ to be

$$O\left(n\frac{((\gamma + \|\mathcal{H}\|_{\text{pauli}})T)^{1.5}}{\sqrt{\epsilon}}\right) \tag{17}$$

one- and two-qubit gates along with

$$\tilde{O}[\log(T/\epsilon)] \tag{18}$$

ancilla if choosing to use the Hamiltonian simulation subroutine of Ref. [7].

For the example of local depolarizing dissipation, consider the case when $\mathcal{D}$ generates one-local depolarizing noise. In this case, the jump operators have the form $L_\mu = \sqrt{\frac{\gamma}{4}}V_\mu$ where $V_\mu \in \{X, Y, Z\}$ and acts on the μth qubit. The analysis follows similar to the above with γ replaced by $n\gamma/4$ in the final results. Similarly, oracles $\mathcal{A}_{\delta t}^{(i)}$ become n-fold tensor products of single-qubit unitaries.

B. Crosstalk in superconducting transmons

As a second demonstration of our algorithm we will simulate the evolution of two superconducting transmon qubits undergoing ZZ-crosstalk noise. ZZ crosstalk is a common source of noise in superconducting transmon qubits given that the two-local Z interaction is always present in the joint system's Hamiltonian. In order to model ZZ-crosstalk on superconducting hardware, Tripathi *et al.* in Ref. [32] use a Lindbladian model with the following system Hamiltonian:

$$H = \sum_{i=1}^{2}\frac{\omega_i}{2}Z_i + JZ_1Z_2, \tag{19}$$

where ω_i is the ith transmon qubit's frequency in the rotating frame and Z_i is the Pauli Z operator acting on the ith transmon qubit. The dissipator $\mathcal{D}$ used has $m = 3$ jump operators of the following form:

$$L_\mu = \begin{cases}\sqrt{\gamma_\mu}Z_\mu & \text{for } i \in \{1,2\}\\ \sqrt{\gamma_\mu}Z_1Z_2 & \text{otherwise}\end{cases}, \tag{20}$$

where $\gamma_\mu \in \mathbb{R}_+ \setminus \{0\}$. The full system dynamics is given by the Lindbladian

$$\begin{aligned}\mathcal{L}(\rho) &= \mathcal{H}(\rho) + \mathcal{D}(\rho)\\ &= -i\left[\sum_{i=1}^{2}\frac{\omega_i}{2} + JZ_1Z_2, \rho\right]\\ &\quad + \sum_{\mu=1}^{2}\gamma_\mu(Z_\mu\rho Z_\mu - \rho) + \gamma_3(Z_1Z_2\rho Z_1Z_2 - \rho). \end{aligned} \tag{21}$$

Since each L_μ is proportional to some unitary, we can approximate the channel $e^{\delta t\mathcal{D}}$ up to arbitrary precision with a stochastically simulatable channel, as shown in Appendix A. Setting this channel to be our dissipative simulation subroutine $\mathcal{N}_{\delta t}$ gives us oracles $\mathcal{A}_{\delta t}^{(i)} \in \{Z_1, Z_2, Z_1Z_2\}$ which can be implemented in the worst case with only two one-qubit gates. Computing $\|\mathcal{L}\|_{\text{pauli}}$ we find

$$\begin{aligned}\|\mathcal{L}\|_{\text{pauli}} &= \sum_{k=0}^{s-1}\beta_{0k} + \sum_{\mu=1}^{m}\left(\sum_{k=0}^{s-1}\beta_{\mu k}\right)^2\\ &= \sum_{i=1}^{2}\frac{\omega_i}{2} + J + \sum_{\mu=1}^{3}\gamma_\mu. \end{aligned} \tag{22}$$

Applying Theorem 1 gives us a total query cost of

$$r = O\left(\left(\left[\sum_{i=1}^{2}\frac{\omega_i}{2} + \sum_{\mu=1}^{3}\gamma_\mu + J\right]T\right)^{1.5} / \sqrt{\epsilon}\right) \tag{23}$$

to both our Hamiltonian simulation subroutine $\mathcal{K}_{\delta t}$ and our dissipative simulation subroutine $\mathcal{N}_{\delta t}$.

Given that all of the terms in the system Hamiltonian H commute, we will choose to instantiate $\mathcal{K}_{\delta t}$ using a product formula algorithm. Applying Theorem 6 of Ref. [29] we find that even a first-order product formula suffices to provide us with an approximation to $e^{\frac{\delta t}{2}\mathcal{H}}$ with no error. Thus we only need to implement an approximation to the unitary $e^{-i\frac{J\delta t}{2}Z_1Z_2}e^{-i\frac{\omega_2\delta t}{2}Z_2}e^{-i\frac{\omega_1\delta t}{2}Z_1}$ where $\delta t = T/r$, this can be done in $O(1)$ one- and two-qubit gates. Thus the total one- and two-qubit gate cost in the worst case to simulate the dynamics of two superconducting transmons undergoing ZZ crosstalk is given by Eq. (23) as well.

C. $1/f$ dephasing in superconducting transmons

As a final demonstration of our algorithm we will describe a simulation of n superconducting transmon qubits undergoing local $1/f$ dephasing. As described in Ref. [33], and in the previous example, the time-independent Hamiltonian for n transmon qubits can be modeled as $H = \sum_{i=1}^{n} H_i = \sum_{i=1}^{n}\omega_i Z_i/2$ where ω_i is the ith transmon's frequency, and Z_i the Pauli Z operator acting on the ith transmon qubit. For this example we assume the ZZ crosstalk between the various transmon qubits has been suppressed and we can drop the two-local interaction terms in the system Hamiltonian. Local dephasing is generated by a dissipator with jump operators of the form $L_\mu(t) = \sqrt{\frac{\Gamma_\mu(t)}{2}}Z_\mu$ where Z_μ is the Pauli Z operator, which acts on the μth transmon qubit. $\Gamma_\mu(t)$ is the dephasing rate for the μth qubit and sources the noise in the system. Since our noise source has a power spectral density of $1/f$, we assume each $\Gamma_\mu(t)$ is a stochastic process with such a power spectral density associated to it. We also assume that each $\Gamma_\mu(t)$ is independent. The dynamics of the system is then given by

$$\begin{aligned}\frac{d\rho}{dt} &= \sum_{\mu=1}^{n}\left(-i\left[\frac{\omega_\mu}{2}Z_\mu, \rho\right] + \frac{\Gamma_\mu(t)}{2}(Z_\mu\rho Z_\mu - \rho)\right)\\ &= \sum_{\mu=1}^{n}(-i[H_\mu, \rho] + \mathcal{D}_\mu(t)(\rho)).\end{aligned} \tag{24}$$

We can sample from the stochastic process $\Gamma_\mu(t)$ that has a power spectral density of approximately $1/f$ by implementing the following procedure described in Ref. [34]. Roughly speaking this procedure involves numerically approximating the stochastic process in frequency space, then inverse Fourier transforming the result. More specifically, given the final time T and number of samples N needed of the random process, we can generate a set of $\{\Gamma_\mu(t_k)\}_{k=0}^{N-1}$ samples where $t_k = k\delta t = Tk/N$ by following the subsequently described procedure. For each frequency $\omega_m = 2\pi m/T$ we sample $X_m, Y_m \sim \mathcal{N}(0, 1)$ where $\mathcal{N}$ denotes the Gaussian distribution. We then compute the mth Fourier weight as

$$\tilde{\Gamma}_\mu(\omega_m) = \sqrt{\frac{1}{2|\omega_m|}}(X_m + iY_m). \tag{25}$$

To assure $\Gamma_\mu(t_k) \in \mathbb{R}$ we constrain $\tilde{\Gamma}_\mu(\omega_{N-m}) = \tilde{\Gamma}_\mu(\omega_m)^*$ allowing us to only compute half of the Fourier weights. Finally we compute each $\Gamma_\mu(t_k)$ as the inverse Fourier transform of $\tilde{\Gamma}_\mu(\omega_m)$ given by

$$\Gamma_\mu(t_k) = \frac{1}{\sqrt{N}}\sum_{m=0}^{N-1}\tilde{\Gamma}_\mu(\omega_m)e^{-i\omega_m t_k}. \tag{26}$$

The resulting set of samples $\{\Gamma_\mu(t_k)\}_{k=1}^{N-1}$ has a power spectral density of approximately $1/f$. We can then take the absolute value of each of the samples giving us a set $\{\Gamma_\mu(t_k)\}_{k=0}^{N-1}$ of dephasing rates, which are positive.

To simulate the dynamics described by Eq. (24) we design subroutines $\mathcal{K}_{\delta t}$, and $\mathcal{N}_{t,\delta t}$ to simulate the coherent and dissipative dynamics respectively. Since each of the superoperators in the sum of Eq. (24) act only on a single qubit, it suffices to consider subroutines $\mathcal{K}_{\delta t}^{(\mu)}$ and $\mathcal{N}_{t,\delta t}^{(\mu)}$ where each subroutine acts only on the μth qubit and

$$\mathcal{K}_{\delta t} = \bigotimes_{\mu=1}^{n}\mathcal{K}_{\delta t}^{(\mu)} \quad \text{and} \quad \mathcal{N}_{t,\delta t} = \bigotimes_{\mu=1}^{n}\mathcal{N}_{t,\delta t}^{(\mu)}. \tag{27}$$

The μth dissipative subroutine approximates the map

$$\begin{aligned}\mathcal{T}e^{\int_{t_i}^{t_i+\delta t} dt\,\mathcal{D}_\mu(t)} &= (1 - \Gamma_\mu(t_i)\delta t + \Gamma_\mu(t_i)\delta t^2)\mathcal{I}\\ &\quad + (\Gamma_\mu(t_i)\delta t - \Gamma_\mu(t_i)\delta t^2)\mathcal{Z}_\mu + \mathcal{O}(\delta t^3),\end{aligned} \tag{28}$$

where $\mathcal{Z}_\mu(\rho) = Z_\mu\rho Z_\mu$. Note, the right-hand side of the above equation fits our definition of being an approximately stochastically simulatable up to error $\mathcal{O}(\delta t^3)$. Thus we define the μth dissipative subroutine as

$$\mathcal{N}_{t_i,\delta t}^{(\mu)} = \{1 - p[\Gamma_\mu(t_i), \delta t]\}\mathcal{I} + p[\Gamma_\mu(t_i), \delta t]\mathcal{Z}_\mu, \tag{29}$$

where probability $p[\Gamma_\mu(t_i), \delta t] = \Gamma_\mu(t_i)\delta t - \Gamma_\mu(t_i)\delta t^2$. Considering the non-Markovian nature of our noise source $\Gamma_\mu(t)$, the probabilities at each time step $p[\Gamma_\mu(t_i), \delta t]$ are correlated with the probabilities at all the other time steps of the simulation. To sample from such a joint distribution, we fix the number of time steps r of the simulation, which implies $\delta t = T/r$. Next, during compilation of the full quantum circuit, we classically sample for each qubit a set of $1/f$ correlated dephasing rates $\{\Gamma_\mu(t_i)\}_{i=1}^{r}$ using the procedure described above. Finally, when compiling each $\mathcal{N}_{t_i,\delta t}^{(\mu)}$ subroutine, we classically compute the probability $p[\Gamma_\mu(t_i), \delta t]$ and sample whether to apply $\mathcal{Z}_\mu$ or not. Each $\mathcal{Z}_\mu$ operation is a single Z gate acting on the μth qubit, thus in the worst case each $\mathcal{N}_{t_i,\delta t}$ subroutine requires $O(n)$ one-qubit gates to implement.

To implement the $\mathcal{K}_{\delta t}$, we note that each $\mathcal{K}_{\delta t}^{(\mu)}$ subroutine approximates the map generated by evolving the μth qubit under the Hamiltonian $H_\mu = \omega_\mu Z_\mu/2$ for time $\delta t/2$. This corresponds to applying the unitary $e^{-i\frac{\omega_\mu\delta t}{4}Z_\mu}$ on the μth qubit, which amounts to applying a single-qubit Z rotation. Thus each $\mathcal{K}_{\delta t}$ subroutine can be implemented with $O(n)$ one-qubit gates. Thus in the worst case the total simulation requires a circuit of $O(rn)$ one-qubit gates in order to simulate the dynamics described by Eq. (24), where r denotes the number of time steps of the simulation.

VI. DISCUSSION

We have presented a novel quantum algorithm for simulating restricted, but physically motivated, types of dynamics generated by Lindblad master equations. Our algorithm leverages sampling in the compilation phase of the quantum circuit to circumvent the overhead of deterministically implementing all of the jump operators needed to simulate the dynamics. Our approach eliminates the explicit scaling on the number of jump operators as it appears in previous algorithms, with the only scaling depending on the norm of the Lindbladian superoperator, which all other Lindbladian simulation algorithms depend on. This strategy also benefits from reduced ancillary qubit costs, given the only ancilla requirements come from the choice of Hamiltonian simulation subroutine. We also treat the closed system dynamics on a different footing than the dissipative dynamics. This approach allows us to leverage the wealth of available closed-system quantum algorithms in addition to extending the algorithm to deal with scenarios where dissipation may be time dependent or correlated over time.

It would be interesting to see if our approach of leveraging classical randomness and treating the dissipative evolution differently than the closed system evolution can be used to achieve better scaling with respect to T and ϵ, ideally $O[T\,\mathrm{polylog}(\frac{1}{\epsilon})]$ of Ref. [20] but without the dependence on the number of jump operators and using only a few ancillary qubits. Extending the algorithm to incorporate more general non-Markovian dynamics is another interesting future direction.

ACKNOWLEDGMENTS

This work is supported by Sandia National Laboratories' Laboratory Directed Research and Development program (Contract No. 2534192). Additional support by DOE's Express: 2023 Exploratory Research For Extreme-scale Science Program under Award No. DE-SC0024685 is acknowledged.

APPENDIX A: EVOLUTION GENERATED BY UNITARY JUMP OPERATORS

Let $\mathcal{D}$ be a Lindbladian dissipator with jump operators L_μ. Setting $L_\mu = \alpha_\mu U_\mu$ where U_μ is unitary gives us $\mathcal{D}(\rho) = \sum_{\mu=1}^{m}(L_\mu \rho L_\mu^\dagger - \frac{1}{2}\{L_\mu^\dagger L_\mu, \rho\}) = \sum_{\mu=1}^{m} |\alpha_\mu|^2 U_\mu \rho U_\mu^\dagger - \sum_{\mu=1}^{m} |\alpha_\mu|^2 \rho$. Define the random unitary channel $\mathcal{R}(\rho) = \sum_{\mu=1}^{m} |\alpha_\mu|^2 U_\mu \rho U_\mu^\dagger$ and $a = \sum_{\mu=1}^{m} |\alpha_\mu|^2$, hence $\mathcal{D}(\rho) = \mathcal{R} - a\mathcal{I}$. Note that $e^{\delta t \mathcal{D}} = e^{-a\delta t \mathcal{I}} e^{\delta t \mathcal{R}}$. Assuming $\delta t \|\mathcal{R}\|_\diamond \leqslant 1$, we have

$$\left\| e^{\delta t \mathcal{D}} - e^{-a\delta t} \sum_{k=0}^{K} \frac{\delta t^k}{k!} \mathcal{R}^k(\rho) \right\|_\diamond \leqslant c e^{-a\delta t} \frac{(\delta t \|\mathcal{R}\|_\diamond)^{K+1}}{(K+1)!}. \tag{A1}$$

Clearly $\sum_{k=0}^{K} e^{-a\delta t} \frac{\delta t^k}{k!} \mathcal{R}^k(\rho)$ is a random unitary channel with non-negative coefficients for any K. Therefore the evolution can be approximated with a random unitary channel up to arbitrary accuracy by increasing K. Note that the sum can involve terms that are up to K multiplication of unitary operators $\{U_\mu\}$. When $\{U_\mu\}$ belongs to a group of efficiently implementable unitaries, such as n-fold tensor products of single-qubit unitaries or Clifford circuits, which are closed under multiplication, the complexity of implementing the resulting unitary operators does not grow with K.

APPENDIX B: TIME-DEPENDENT SECOND-ORDER PRODUCT FORMULA

In this section, we prove a time-dependent second-order product formula bound for Liouvillians. For simplicity, we assume that the Liouvillian $\mathcal{L}$ splits up into a time-independent component $\mathcal{H}$ and a time-dependent component $\mathcal{D}(t)$. This bound is used in the main text when generalizing our algorithm to incorporate time-dependent dissipators.

Before proving the main result we cite an important Lemma from Kliesch *et al.* [12] that will be useful in the proof.

Lemma 1. Backward time evolution [12]. Let $\mathcal{L}$ be a Liouvillian. For $t > s$:

(i) $T_{\mathcal{L}}(t, s) = \mathcal{T} e^{\int_s^t \mathcal{L}(t')dt'}$ is invertible and its inverse is $T_{\mathcal{L}}^{-}(t, s)$ as defined by

$$\partial_t T_{\mathcal{L}}^{-}(t, s) = -T_{\mathcal{L}}^{-}(t, s)\mathcal{L}. \tag{B1}$$

(ii) If the Liouvillian $\mathcal{L}$ is piecewise continuous in time then

$$\|T_{\mathcal{L}}^{-}(t, s)\|_\diamond \leqslant e^{\int_s^t \|\mathcal{L}(t')\|_\diamond dt'}. \tag{B2}$$

■

Proof. See Ref. [12].

We now prove the main result.

Theorem 2 (restated). Let $\mathcal{H}$ and $\mathcal{D}(t)$ be Liouvillians and $0 \leqslant (\frac{\|\mathcal{H}\|_\diamond}{2} + \sup_t \|\mathcal{D}(t)\|_\diamond)(t - s) \leqslant 1$, then

$$\|\mathcal{T} e^{\int_s^t (\mathcal{H}+\mathcal{D}(t'))dt'} - e^{\frac{(t-s)}{2}\mathcal{H}} \circ \mathcal{T} e^{\int_s^t \mathcal{D}(t')dt'} \circ e^{\frac{(t-s)}{2}\mathcal{H}}\|_\diamond \leqslant \frac{\sup_t \|[\mathcal{H}, \mathcal{D}(t)]\|_\diamond}{3} \left(\frac{\|\mathcal{H}\|_\diamond}{2} + \sup_t \|\mathcal{D}(t)\|_\diamond \right) (t - s)^3. \tag{B3}$$

Proof. Define $T_{\mathcal{L}}(t,s)=\mathcal{T}e^{\int_s^t \mathcal{L}(t')dt'}$ and $T^-_{\mathcal{L}}(t,s)$ according to Lemma 1. Both $T_{\mathcal{L}}(t,s)$ and $T^-_{\mathcal{L}}(t,s)$ are the unique solutions to the differential equations

$$\partial_t T_{\mathcal{L}}(t,s)=\mathcal{L}(t)T_{\mathcal{L}}(t,s) \quad \text{and} \quad \partial_t T^-_{\mathcal{L}}(t,s)=-T^-_{\mathcal{L}}(t,s)\mathcal{L}(t) \tag{B4}$$

with the initial condition $T_{\mathcal{L}}(s,s)=T^-_{\mathcal{L}}(s,s)=\mathcal{I}$. Rewriting the left-hand side of equation (B3) in terms of our definitions gives

$$\epsilon=\left\|T_{\mathcal{H}+\mathcal{D}}(t,s)-T_{\mathcal{H}}\left(t,\frac{t+s}{2}\right)T_{\mathcal{D}}(t,s)T_{\mathcal{H}}\left(\frac{t+s}{2},s\right)\right\|_\diamond, \tag{B5}$$

which can be rewritten as

$$\begin{aligned}\epsilon&=\left\|T_{\mathcal{H}}\left(t,\frac{t+s}{2}\right)T_{\mathcal{D}}(t,s)T_{\mathcal{H}}\left(\frac{t+s}{2},s\right)\left(T^-_{\mathcal{H}}\left(\frac{t+s}{2},s\right)T^-_{\mathcal{D}}(t,s)T^-_{\mathcal{H}}\left(t,\frac{t+s}{2}\right)T_{\mathcal{H}+\mathcal{D}}(t,s)-\mathcal{I}\right)\right\|_\diamond &\text{(B6)}\\ &=\left\|T_{\mathcal{H}}\left(t,\frac{t+s}{2}\right)T_{\mathcal{D}}(t,s)T_{\mathcal{H}}\left(\frac{t+s}{2},s\right)F(t,s)\right\|_\diamond, &\text{(B7)}\end{aligned}$$

where we have defined $F(t,s)=T^-_{\mathcal{H}}(\frac{t+s}{2},s)T^-_{\mathcal{D}}(t,s)T^-_{\mathcal{H}}(t,\frac{t+s}{2})T_{\mathcal{H}+\mathcal{D}}(t,s)-\mathcal{I}$. Focusing on $F(t,s)$ and using the fundamental theorem of calculus one finds:

$$\begin{aligned}F(t,s)&=\int_s^t \partial_r\left(T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)T_{\mathcal{H}+\mathcal{D}}(r,s)\right)dr\\ &=\int_s^t\Bigg\{\partial_r\left[T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\right]T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)T_{\mathcal{H}+\mathcal{D}}(r,s)+T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\partial_r[T^-_{\mathcal{D}}(r,s)]T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)T_{\mathcal{H}+\mathcal{D}}(r,s)\\ &\quad+T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)T^-_{\mathcal{D}}(r,s)\partial_r\left[T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)\right]T_{\mathcal{H}+\mathcal{D}}(r,s)+T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)\partial_r[T_{\mathcal{H}+\mathcal{D}}(r,s)]\Bigg\}dr.\end{aligned} \tag{B8}$$

Applying identities: $\partial_t T_{\mathcal{H}+\mathcal{D}}(t,s)=(\mathcal{H}+\mathcal{D}(t))T_{\mathcal{H}+\mathcal{D}}(t,s)$, $\partial_t T^-_{\mathcal{H}}(\frac{t+s}{2},s)=-\frac{1}{2}T^-_{\mathcal{H}}(\frac{t+s}{2},s)\mathcal{H}$, $\partial_t T^-_{\mathcal{H}}(t,\frac{t+s}{2})=-\frac{1}{2}T^-_{\mathcal{H}}(t,\frac{t+s}{2})\mathcal{H}$, and $\partial_t T^-_{\mathcal{D}}(t,s)=-T^-_{\mathcal{D}}(t,s)\mathcal{D}(t)$ yields,

$$\begin{aligned}F(t,s)&=\int_s^t T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\Bigg\{-\frac{\mathcal{H}}{2}T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)-T^-_{\mathcal{D}}(r,s)\mathcal{D}(r)T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)-T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)\frac{\mathcal{H}}{2}\\ &\quad+T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)(\mathcal{H}+\mathcal{D}(r))\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dr\\ &=\int_s^t T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\Bigg\{-\frac{\mathcal{H}}{2}T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)-T^-_{\mathcal{D}}(r,s)\mathcal{D}(r)T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\\ &\quad+T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(r)\right)\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dr\\ &=-\int_s^t T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\Bigg\{\left[T^-_{\mathcal{D}}(r,s),\mathcal{D}(r)\right]T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)+\left[\frac{\mathcal{H}}{2}+\mathcal{D}(r),T^-_{\mathcal{D}}(r,s)T^-_{\mathcal{H}}\left(r,\frac{r+s}{2}\right)\right]\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dr.\end{aligned} \tag{B9}$$

Applying the fundamental theorem of calculus to the terms inside each of the commutators and using the above derivative identities again yields,

$$\begin{aligned}F(t,s)&=-\int_s^t T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\int_s^r\Bigg\{\partial_u\left([T^-_{\mathcal{D}}(u,s),\mathcal{D}(r)]T^-_{\mathcal{H}}\left(\frac{u+s}{2},s\right)\right)\\ &\quad+\partial_u\left(\left[\frac{\mathcal{H}}{2}+\mathcal{D}(r),T^-_{\mathcal{D}}(u,s)T^-_{\mathcal{H}}\left(u,\frac{u+s}{2}\right)\right]\right)\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr\\ &=-\int_s^t T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\int_s^r\Bigg\{[\partial_u[T^-_{\mathcal{D}}(u,s)],\mathcal{D}(r)]T^-_{\mathcal{H}}\left(\frac{u+s}{2},s\right)\\ &\quad+[T^-_{\mathcal{D}}(u,s),\mathcal{D}(r)]\partial_u\left[T^-_{\mathcal{H}}\left(\frac{u+s}{2},s\right)\right]\\ &\quad+\left[\frac{\mathcal{H}}{2}+\mathcal{D}(r),\partial_u[T^-_{\mathcal{D}}(u,s)]T^-_{\mathcal{H}}\left(u,\frac{u+s}{2}\right)+T^-_{\mathcal{D}}(u,s)\partial_u\left[T^-_{\mathcal{H}}\left(u,\frac{u+s}{2}\right)\right]\right]\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr\\ &=\int_s^t T^-_{\mathcal{H}}\left(\frac{r+s}{2},s\right)\int_s^r\Bigg\{[T^-_{\mathcal{D}}(u,s)\mathcal{D}(u),\mathcal{D}(r)]T^-_{\mathcal{H}}\left(\frac{u+s}{2},s\right)+[T^-_{\mathcal{D}}(u,s),\mathcal{D}(r)]T^-_{\mathcal{H}}\left(\frac{u+s}{2},s\right)\frac{\mathcal{H}}{2}\end{aligned}$$

$$\begin{aligned}
&+\left[\frac{\mathcal{H}}{2}+\mathcal{D}(r), T_{\mathcal{D}}^{-}(u,s)\mathcal{D}(u)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)+T_{\mathcal{D}}^{-}(u,s)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\frac{\mathcal{H}}{2}\right]\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr\\
&=\int_s^t\int_s^r T_{\mathcal{H}}^{-}\left(\frac{r+s}{2},s\right)\Bigg\{T_{\mathcal{D}}^{-}(u,s)[\mathcal{D}(u),\mathcal{D}(r)]T_{\mathcal{H}}^{-}\left(\frac{u+s}{2},s\right)+[T_{\mathcal{D}}^{-}(u,s),\mathcal{D}(r)]\mathcal{D}(u)T_{\mathcal{H}}^{-}\left(\frac{u+s}{2},s\right)\\
&\quad+[T_{\mathcal{D}}^{-}(u,s),\mathcal{D}(r)]\frac{\mathcal{H}}{2}T_{\mathcal{H}}^{-}\left(\frac{u+s}{2},s\right)+\left[\frac{\mathcal{H}}{2}+\mathcal{D}(r),T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\right]\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr,
\end{aligned}\tag{B10}$$

where the last equality follows from commuting $T_{\mathcal{H}}^{-}$ and $\mathcal{H}$ since $\mathcal{H}$ is time independent. Massaging the last commutator in the sum yields,

$$\begin{aligned}
F(t,s)&=\int_s^t\int_s^r T_{\mathcal{H}}^{-}\left(\frac{r+s}{2},s\right)\Bigg\{T_{\mathcal{D}}^{-}(u,s)[\mathcal{D}(u),\mathcal{D}(r)]T_{\mathcal{H}}^{-}\left(\frac{u+s}{2},s\right)+[T_{\mathcal{D}}^{-}(u,s),\mathcal{D}(r)]\mathcal{D}(u)T_{\mathcal{H}}^{-}\left(\frac{u+s}{2},s\right)\\
&\quad+[T_{\mathcal{D}}^{-}(u,s),\mathcal{D}(r)]\frac{\mathcal{H}}{2}T_{\mathcal{H}}^{-}\left(\frac{u+s}{2},s\right)+T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\left[\mathcal{D}(r),T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\right]\\
&\quad+T_{\mathcal{D}}^{-}(u,s)\left[\frac{\mathcal{H}}{2},\mathcal{D}(u)\right]T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)+T_{\mathcal{D}}^{-}(u,s)\left[\mathcal{D}(r),\frac{\mathcal{H}}{2}\right]T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\\
&\quad+T_{\mathcal{D}}^{-}(u,s)[\mathcal{D}(r),\mathcal{D}(u)]T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)+\left[\frac{\mathcal{H}}{2},T_{\mathcal{D}}^{-}(u,s)\right]\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\\
&\quad+[\mathcal{D}(r),T_{\mathcal{D}}^{-}(u,s)]\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr.
\end{aligned}\tag{B11}$$

Simplifying the equation gives us,

$$\begin{aligned}
F(t,s)&=\int_s^t\int_s^r T_{\mathcal{H}}^{-}\left(\frac{r+s}{2},s\right)\Bigg\{-T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\left[T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right),\mathcal{D}(r)\right]\\
&\quad-\left[T_{\mathcal{D}}^{-}(u,s),\frac{\mathcal{H}}{2}\right]\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)+T_{\mathcal{D}}^{-}(u,s)\left[\frac{\mathcal{H}}{2},\mathcal{D}(u)-\mathcal{D}(r)\right]T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr\\
&=\int_s^t\int_s^r T_{\mathcal{H}}^{-}\left(\frac{r+s}{2},s\right)\Bigg\{-T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\left(\int_s^u\partial_v\left[T_{\mathcal{H}}^{-}\left(v,\frac{v+s}{2}\right)\mathcal{D}(r)T_{\mathcal{H}}\left(v,\frac{v+s}{2}\right)\right]dv\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\\
&\quad-\left(\int_s^u\partial_v\left[T_{\mathcal{D}}^{-}(v,s)\frac{\mathcal{H}}{2}T_{\mathcal{D}}(v,s)\right]dv\right)T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\\
&\quad+T_{\mathcal{D}}^{-}(u,s)\left[\frac{\mathcal{H}}{2},\int_r^u\partial_v\mathcal{D}(v)dv\right]T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr.
\end{aligned}\tag{B12}$$

Applying the fundamental theorem of calculus again yields

$$\begin{aligned}
F(t,s)&=\int_s^t\int_s^r T_{\mathcal{H}}^{-}\left(\frac{r+s}{2},s\right)\Bigg\{-T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\left(\int_s^u\partial_v\left[T_{\mathcal{H}}^{-}\left(v,\frac{v+s}{2}\right)\mathcal{D}(r)T_{\mathcal{H}}\left(v,\frac{v+s}{2}\right)\right]dv\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\\
&\quad-\left(\int_s^u\partial_v\left[T_{\mathcal{D}}^{-}(v,s)\frac{\mathcal{H}}{2}T_{\mathcal{D}}(v,s)\right]dv\right)T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\\
&\quad+T_{\mathcal{D}}^{-}(u,s)\left[\frac{\mathcal{H}}{2},\int_r^u\partial_v\mathcal{D}(v)dv\right]T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr.
\end{aligned}\tag{B13}$$

Continuing to simplify our expression for $F(t,s)$ gives us

$$\begin{aligned}
F(t,s)&=\int_s^t\int_s^r T_{\mathcal{H}}^{-}\left(\frac{r+s}{2},s\right)\Bigg\{\int_s^u T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(v,\frac{v+s}{2}\right)\left[\frac{\mathcal{H}}{2},\mathcal{D}(r)\right]T_{\mathcal{H}}\left(v,\frac{v+s}{2}\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)dv\\
&\quad+\int_s^u T_{\mathcal{D}}^{-}(v,s)\left[\mathcal{D}(v),\frac{\mathcal{H}}{2}\right]T_{\mathcal{D}}(v,s)T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)dv\\
&\quad+\int_r^u T_{\mathcal{D}}^{-}(u,s)\left[\frac{\mathcal{H}}{2},\partial_v\mathcal{D}(v)\right]T_{\mathcal{H}}^{-}\left(u,\frac{u+s}{2}\right)dv\Bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr\\
&=\int_s^t\int_s^r T_{\mathcal{H}}^{-}\left(\frac{r+s}{2},s\right)\Bigg\{\int_s^u T_{\mathcal{D}}^{-}(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^{-}\left(v,\frac{v+s}{2}\right)\left[\frac{\mathcal{H}}{2},\mathcal{D}(r)\right]T_{\mathcal{H}}^{-}\left(\frac{u+s}{2},\frac{v+s}{2}\right)dv
\end{aligned}$$

$$\begin{aligned}
&+\int_s^u T_{\mathcal{D}}^-(v,s)\left[\mathcal{D}(v),\frac{\mathcal{H}}{2}\right]T_{\mathcal{D}}^-(u,v)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^-\left(u,\frac{u+s}{2}\right)dv\\
&+\int_r^u T_{\mathcal{D}}^-(u,s)\left[\frac{\mathcal{H}}{2},\partial_v\mathcal{D}(v)\right]T_{\mathcal{H}}^-\left(u,\frac{u+s}{2}\right)dv\bigg\}T_{\mathcal{H}+\mathcal{D}}(r,s)dudr.
\end{aligned}\tag{B14}$$

Returning to Eq. (B6) we find,

$$\begin{aligned}
\epsilon \leqslant{}& \int_s^t\int_s^r \left\|T_{\mathcal{H}}\left(t,\frac{t+s}{2}\right)T_{\mathcal{D}}(t,s)T_{\mathcal{H}}\left(\frac{t+s}{2},\frac{r+s}{2}\right)\right\|_\diamond\\
&\times\bigg\{\int_s^u \left\|T_{\mathcal{D}}^-(u,s)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^-\left(v,\frac{v+s}{2}\right)\left[\frac{\mathcal{H}}{2},\mathcal{D}(r)\right]T_{\mathcal{H}}^-\left(\frac{u+s}{2},\frac{v+s}{2}\right)\right\|_\diamond dv\\
&+\int_s^u \left\|T_{\mathcal{D}}^-(v,s)\left[\mathcal{D}(v),\frac{\mathcal{H}}{2}\right]T_{\mathcal{D}}^-(u,v)\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)T_{\mathcal{H}}^-\left(u,\frac{u+s}{2}\right)\right\|_\diamond dv\\
&+\int_r^u \left\|T_{\mathcal{D}}^-(u,s)\left[\frac{\mathcal{H}}{2},\partial_v\mathcal{D}(v)\right]T_{\mathcal{H}}^-\left(u,\frac{u+s}{2}\right)\right\|_\diamond dv\bigg\}\|T_{\mathcal{H}+\mathcal{D}}(r,s)\|_\diamond dudr\\
\leqslant{}& \int_s^t\int_s^r\bigg\{\int_s^u \|T_{\mathcal{D}}^-(u,s)\|_\diamond\left\|\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\right\|_\diamond\left\|T_{\mathcal{H}}^-\left(v,\frac{v+s}{2}\right)\right\|_\diamond\left\|\left[\frac{\mathcal{H}}{2},\mathcal{D}(r)\right]\right\|_\diamond\left\|T_{\mathcal{H}}^-\left(\frac{u+s}{2},\frac{v+s}{2}\right)\right\|_\diamond dv\\
&+\int_s^u \|T_{\mathcal{D}}^-(v,s)\|_\diamond\left\|\left[\mathcal{D}(v),\frac{\mathcal{H}}{2}\right]\right\|_\diamond\|T_{\mathcal{D}}^-(u,v)\|_\diamond\left\|\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\right\|_\diamond\left\|T_{\mathcal{H}}^-\left(u,\frac{u+s}{2}\right)\right\|_\diamond dv\\
&+\int_r^u \|T_{\mathcal{D}}^-(u,s)\|_\diamond\left\|\left[\frac{\mathcal{H}}{2},\partial_v\mathcal{D}(v)\right]\right\|_\diamond\left\|T_{\mathcal{H}}^-\left(u,\frac{u+s}{2}\right)\right\|_\diamond dv\bigg\}dudr,
\end{aligned}\tag{B15}$$

where the second inequality follows from $\|T_{\mathcal{L}}(t,s)\|_\diamond = 1$ since $T_{\mathcal{L}}(t,s)$ is a CPTP map. Applying part 2 of Lemma 1 yields,

$$\begin{aligned}
\epsilon \leqslant{}& \int_s^t\int_s^r\bigg\{\int_s^u e^{\frac{\|\mathcal{H}\|_\diamond}{2}(u-s)+\int_s^u\|\mathcal{D}(t')\|_\diamond dt'}\left\|\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\right\|_\diamond\left\|\left[\frac{\mathcal{H}}{2},\mathcal{D}(r)\right]\right\|_\diamond dv\\
&+\int_s^u e^{\frac{\|\mathcal{H}\|_\diamond}{2}(u-s)+\int_s^u\|\mathcal{D}(t')\|_\diamond dt'}\left\|\left[\mathcal{D}(v),\frac{\mathcal{H}}{2}\right]\right\|_\diamond\left\|\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\right\|_\diamond dv\\
&+\int_r^u e^{\frac{\|\mathcal{H}\|_\diamond}{2}(u-s)+\int_s^u\|\mathcal{D}(t')\|_\diamond dt'}\left\|\left[\frac{\mathcal{H}}{2},\partial_v\mathcal{D}(v)\right]\right\|_\diamond dv\bigg\}dudr\\
\leqslant{}& \int_s^t\int_s^r e^{(\frac{\|\mathcal{H}\|_\diamond}{2}+\sup_t\|\mathcal{D}(t)\|_\diamond)(u-s)}\bigg\{\int_s^u\left\|\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\right\|_\diamond\left\|\left[\frac{\mathcal{H}}{2},\mathcal{D}(r)\right]\right\|_\diamond dv\\
&+\int_s^u\left\|\left[\mathcal{D}(v),\frac{\mathcal{H}}{2}\right]\right\|_\diamond\left\|\left(\frac{\mathcal{H}}{2}+\mathcal{D}(u)\right)\right\|_\diamond dv+\int_r^u\left\|\left[\frac{\mathcal{H}}{2},\partial_v\mathcal{D}(v)\right]\right\|_\diamond dv\bigg\}dudr\\
\leqslant{}& \int_s^t\int_s^r e^{(\frac{\|\mathcal{H}\|_\diamond}{2}+\sup_t\|\mathcal{D}(t)\|_\diamond)(u-s)}\bigg\{\left(\frac{\|\mathcal{H}\|_\diamond}{2}+\sup_t\|\mathcal{D}(t)\|_\diamond\right)\sup_t\|[\mathcal{H},\mathcal{D}(t)]\|_\diamond(u-s)\\
&+\sup_t\left\|\left[\frac{\mathcal{H}}{2},\partial_t\mathcal{D}(t)\right]\right\|_\diamond((u-s)-(r-s))\bigg\}dudr.
\end{aligned}\tag{B16}$$

Defining $L = \frac{\|\mathcal{H}\|_\diamond}{2} + \sup_t\|\mathcal{D}(t)\|_\diamond$, $C = \sup_t\|[\mathcal{H},\mathcal{D}(t)]\|_\diamond$, and $D = \sup_t\|[\mathcal{H},\partial_t\mathcal{D}(t)]\|_\diamond$ gives us the bound,

$$\begin{aligned}
\epsilon \leqslant{}& \int_s^t\int_s^r e^{L(u-s)}\left\{\frac{2LC+D}{2}(u-s)-\frac{D}{2}(r-s)\right\}dudr\\
\leqslant{}& \int_s^t\left\{\frac{2LC+D}{2L^2}[e^{L(r-s)}(L(r-s)-1)+1]-\frac{D}{2L^2}L(r-s)[e^{L(r-s)}-1]\right\}dr\\
\leqslant{}& \left\{\frac{2LC+D}{2L^3}(2(1-e^{L(t-s)})+L(t-s)(1+e^{L(t-s)}))-\frac{D}{2L^3}\left((L(t-s)-1)e^{L(t-s)}+1-\frac{(L(t-s))^2}{2}\right)\right\}.
\end{aligned}\tag{B17}$$

Rewriting the bound we find,

$$\epsilon \leqslant \left\{ \frac{2LC+D}{2L^3}\left(2(1-e^{L(t-s)}) + L(t-s)\left(1+e^{L(t-s)}\right)\right) - \frac{D}{2L^3}\left((L(t-s)-1)e^{L(t-s)} + 1 - \frac{(L(t-s))^2}{2}\right)\right\}$$

$$\epsilon \leqslant \left\{ \frac{2LC+D}{2L^3}\left(\frac{(L(t-s))^3}{3} + \mathcal{O}((L(t-s))^4)\right) - \frac{D}{2L^3}\left(\frac{(L(t-s))^3}{3} + \mathcal{O}((L(t-s))^4)\right)\right\} \tag{B18}$$

Assuming $(\frac{\|\mathcal{H}\|_\diamond}{2} + \sup_t \|\mathcal{D}(t)\|_\diamond)(t-s) = L(t-s) \leqslant 1$, gives us

$$\epsilon \leqslant \frac{LC}{3}(t-s)^3 = \frac{1}{3}\sup_t \|[\mathcal{H}, \mathcal{D}(t)]\|_\diamond \left(\frac{\|\mathcal{H}\|_\diamond}{2} + \sup_t \|\mathcal{D}(t)\|_\diamond\right)(t-s)^3. \tag{B19}$$

■

[1] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. **21**, 467 (1982).

[2] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, Commun. Math. Phys. **270**, 359 (2007).

[3] D. W. Berry, R. Cleve, and S. Gharibian, Gate-efficient discrete simulations of continuous-time quantum query algorithms, Quantum Info. Comput. **14**, 1 (2014).

[4] D. W. Berry, A. M. Childs, Childs, R. Cleve, R. Kothari, and R. D. Somma, Exponential improvement in precision for simulating sparse Hamiltonians, in *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing* (ACM, New York, 2014).

[5] A. M. Childs, A. Ostrander, and Y. Su, Faster quantum simulation by randomization, Quantum **3**, 182 (2019).

[6] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum **3**, 163 (2019).

[7] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian dynamics with a truncated taylor series, Phys. Rev. Lett. **114**, 090502 (2015).

[8] E. Campbell, Random compiler for fast Hamiltonian simulation, Phys. Rev. Lett. **123**, 070503 (2019).

[9] K. Nakaji, M. Bagherimehrab, and A. Aspuru-Guzik, High-order randomized compiler for Hamiltonian simulation, PRX Quantum **5**, 020330 (2024).

[10] A. M. Childs and T. Li, Efficient simulation of sparse Markovian quantum dynamics, Quantum Info. Comput. **17**, 901 (2017).

[11] G. D. Bartolomeo, M. Vischi, T. Feri, A. Bassi, and S. Donadi, Efficient quantum algorithm to simulate open systems through a single environmental qubit, Phys. Rev. Res. **6**, 043321 (2024).

[12] M. Kliesch, T. Barthel, C. Gogolin, M. Kastoryano, and J. Eisert, Dissipative quantum church-turing theorem, Phys. Rev. Lett. **107**, 120501 (2011).

[13] X. Li and C. Wang, Simulating Markovian open quantum systems using higher-order series expansion, arXiv:2212.02051.

[14] X. Li and C. Wang, Succinct description and efficient simulation of non-markovian open quantum systems, Commun. Math. Phys. **401**, 147 (2023).

[15] A. W. Schlimgen, K. Head-Marsden, L. M. Sager, P. Narang, and D. A. Mazziotti, Quantum simulation of open quantum systems using a unitary decomposition of operators, Phys. Rev. Lett. **127**, 270503 (2021).

[16] A. W. Schlimgen, K. Head-Marsden, L. M. Sager, P. Narang, and D. A. Mazziotti, Quantum simulation of the lindblad equation using a unitary decomposition of operators, Phys. Rev. Res. **4**, 023216 (2022).

[17] A. W. Schlimgen, K. Head-Marsden, L. M. Sager-Smith, P. Narang, and D. A. Mazziotti, Quantum state preparation and nonunitary evolution with diagonal operators, Phys. Rev. A **106**, 022414 (2022).

[18] N. Suri, J. Barreto, S. Hadfield, N. Wiebe, F. Wudarski, and J. Marshall, Two-unitary decomposition algorithm and open quantum system simulation, Quantum **7**, 1002 (2023).

[19] Z. Hu, R. Xia, and S. Kais, A quantum algorithm for evolving open quantum dynamics on quantum computing devices, Sci. Rep. **10**, 3301 (2020).

[20] in *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, Leibniz International Proceedings in Informatics (LIPIcs), edited by I. Chatzigiannakis, P. Indyk, F. Kuhn, and A. Muscholl (Schloss Dagstuhl Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2017), Vol. 80, pp. 17:1–17:14.

[21] J. D. Guimarães, J. Lim, M. I. Vasilevskiy, S. F. Huelga, and M. B. Plenio, Noise-assisted digital quantum simulation of open systems using partial probabilistic error cancellation, PRX Quantum **4**, 040329 (2023).

[22] J. D. Guimarães, A. Ruiz-Molero, J. Lim, M. I. Vasilevskiy, S. F. Huelga, and M. B. Plenio, Optimized noise-assisted simulation of the lindblad equation with time-dependent coefficients on a noisy quantum processor, Phys. Rev. A **109**, 052224 (2024).

[23] J. Joo and T. P. Spiller, Commutation simulator for open quantum dynamics, New J. Phys. **25**, 083041 (2023).

[24] C.-F. Chen, M. J. Kastoryano, F. G. S. L. Brandão, and A. Gilyén, Quantum thermal state preparation, arXiv:2303.18224.

[25] H.-Y. Liu, X. Lin, Z.-Y. Chen, C. Xue, T.-P. Sun, Q.-S. Li, X.-N. Zhuang, Y.-J. Wang, Y.-C. Wu, M. Gong, and G.-P. Guo, Simulation of open quantum systems on universal quantum computers, arXiv:2405.20712.

[26] A. H. Werner, D. Jaschke, P. Silvi, M. Kliesch, T. Calarco, J. Eisert, and S. Montangero, Positive tensor network approach for simulating open quantum many-body systems, Phys. Rev. Lett. **116**, 237201 (2016).

[27] J. Watrous, Basic notions of quantum information, in *The Theory of Quantum Information* (Cambridge University Press, Cambridge, 2018), pp. 58–123.

[28] S. Lloyd, Universal quantum simulators, Science **273**, 1073 (1996).

[29] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, Theory of trotter error with commutator scaling, Phys. Rev. X **11**, 011020 (2021).

[30] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, 2010).

[31] M.-D. Choi, Completely positive linear maps on complex matrices, Linear Algebra Appl. **10**, 285 (1975).

[32] V. Tripathi, H. Chen, M. Khezri, K.-W. Yip, E. M. Levenson-Falk, and D. A. Lidar, Suppression of crosstalk in superconducting qubits using dynamical decoupling, Phys. Rev. Appl. **18**, 024068 (2022).

[33] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, A quantum engineer's guide to superconducting qubits, Appl. Phys. Rev. **6**, 021318 (2019).

[34] E. Milotti, 1/f noise: A pedagogical review, arXiv:physics/0204033 [physics.class-ph].