

Quantum-Trajectory-Inspired Lindbladian Simulation

Sirui Peng^{1,2}, Xiaoming Sun^{1,2}, Qi Zhao³, and Hongyi Zhou^{1,*}

¹State Key Lab of Processors, *Institute of Computing Technology, Chinese Academy of Sciences*, 100190 Beijing, China

²School of Computer Science and Technology, *University of Chinese Academy of Sciences*, Beijing 100049, China

³QICI Quantum Information and Computation Initiative, School of Computing and Data Science, *The University of Hong Kong*, Pokfulam Road, Hong Kong SAR, China



(Received 12 September 2024; revised 18 February 2025; accepted 15 August 2025; published 22 September 2025)

Simulating the dynamics of open quantum systems is a crucial task in quantum computing, offering wide-ranging applications but remaining computationally challenging. In this paper, we propose two quantum algorithms for simulating the dynamics of open quantum systems governed by Lindbladians. We introduce a new approximation channel for short-time evolution, inspired by the quantum trajectory method, which underpins the efficiency of our algorithms. The first algorithm achieves a gate complexity independent of the number of jump operators, m , marking a significant improvement in efficiency. The second algorithm achieves near-optimal dependence on the evolution time t and precision ϵ and introduces only an additional $\tilde{O}(m)$ factor, which strictly improves upon state-of-the-art gate-based quantum algorithm that has an $\tilde{O}(m^2)$ factor. The improvement stems from the integration of the new approximation channel with a novel structured linear combination of unitaries method. In both our algorithms, the reduction of dependence on m significantly enhances the efficiency of simulating practical dissipative processes characterized by a large number of jump operators.

DOI: [10.1103/ssrs-8x32](https://doi.org/10.1103/ssrs-8x32)

I. INTRODUCTION

Simulating physical systems using quantum-mechanics-based computational devices [1] is a foundational motivation for developing quantum computers. Traditionally, quantum algorithms have primarily focused on the Hamiltonian dynamics of closed systems, assuming no interaction with the environment. In contrast, the paradigm of open quantum systems incorporates environmental effects, offering a more realistic depiction of real-world systems, albeit at the cost of increased computational complexity. The simulation of open systems has crucial applications in various fields [2–8], such as quantum sensing [6,7], quantum chemistry [2,3], mathematical optimization [8]. In addition, open-system simulation enables Hamiltonian eigenstate simulation [9,10], including ground-state preparation, which is a central and long-standing challenge in quantum many-body physics.

A widely studied case is the Markovian open quantum system, which can describe many practical quantum systems through operators acting solely on the system itself. The evolution of a Markovian open system is governed by the Lindblad master equation (LME):

$$\begin{aligned} \frac{d\rho(t)}{dt} &= -i[H, \rho(t)] + \sum_{\alpha=1}^m \left(L_{\alpha}\rho(t)L_{\alpha}^{\dagger} - \frac{1}{2} \{L_{\alpha}^{\dagger}L_{\alpha}, \rho(t)\} \right) \\ &= \mathcal{L}\rho(t), \end{aligned} \quad (1)$$

where $\mathcal{L}$ is the Lindbladian, H is the system Hamiltonian, and m jump operators $L_1, \dots, L_m$ representing environmental effects. Our goal is to construct circuits that implement a channel $\mathcal{N}$ such that $\|\mathcal{N} - e^{\mathcal{L}t}\|_{\diamond} \leq \epsilon$.

There has been increasing interest in open quantum system simulation in recent years [11–23]. The first efficient algorithm with exponentially small precision was proposed by Cleve and Wang [15], achieving a gate cost of $O(m^2 t \text{poly log}(mt/\epsilon))$. Li and Wang [16] further explore simulating Lindbladian using higher series expansion, and obtain an efficient algorithm with $O(m t \text{poly log}(mt/\epsilon))$ calls of the block-encoding circuit of the Lindbladian and $O(m^2 t \text{poly log}(mt/\epsilon))$ additional elementary gates. There are also attempts to reduce Lindbladian simulation to

*Contact author: zhouhongyi@ict.ac.cn

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

Hamiltonian simulation [17,18], but these attempts do not achieve the exponential precision dependence. Kato *et al.* [19] focuses on minimizing circuit depth and ancilla size via a random circuit compilation method. Patel and Wilde [20,21] consider the program state input model and analyze the sampling complexity. Guimarães *et al.* [22], Wei *et al.* [23], Wei *et al.* [24] address the problem of open-system simulation in models other than the standard quantum circuit model.

For all methods discussed above, the gate complexity dependence on the number of jump operators m is at least $\Omega(m)$, which can scale exponentially with the qubit number n , leading to a huge overhead. Consider an n -qubit system where dissipation occurs through collective lowering operations involving arbitrary subsets of qubits. Such a scenario is representative of systems where qubits are closely spaced and interact with the environment in complex, collective manners. In this context, the jump operators associated with the dissipation can be defined for every possible subset S of qubits as

$$L_S = \sqrt{\gamma_S} \prod_{i \in S} \sigma_i^-, \quad (2)$$

where S is any nonempty subset of $\{1, 2, \dots, n\}$, γ_S is the collective decay rate for the subset S and $\sigma_i^- = |0\rangle\langle 1|_i$ is the lowering operator for the i th qubit. In such physical systems, the number of jump operators can be as large as $O(4^n)$. Therefore, reducing m dependence can provide exponential speedup for the simulation of such physical systems. Several proposals [25,26] aim to reduce m dependence, restricted to specific classes of Lindbladians. In Ref. [25], it is assumed that the short-time evolution of the Lindbladian can be well approximated by a mixed-unitary channel. In Ref. [26], the jump operators are limited to be tensor products of Pauli operators $P \in \{I, X, Y, Z\}^{\otimes n}$.

To solve the problem above, we propose two novel quantum algorithms for the simulation of general Markovian open quantum systems, greatly reducing the dependence on the number of jump operators. These algorithms are inspired by the quantum trajectory method in classical simulations of open systems (see Fig. 1), which enables us to find a new approximation channel for short-time evolution. We propose a new technique for implementing quantum channels in the quantum circuit model, which we term “structured linear combination of unitaries”. This technique enables the efficient construction of the new approximation channel. As a result, Algorithm 1 eliminates m dependence in gate complexity through a simple sampling method. Algorithm 2 further incorporates the fractional query method [15,27–29], reducing m dependence from $\tilde{O}(m^2)$ to $\tilde{O}(m)$ while maintaining near-optimal (t, ϵ) dependence of $O(t((\log^2(t/\epsilon))/(\log \log(t/\epsilon))))$. Our

algorithms capture the classical randomness in the Lindblad master equation, allowing the simulation of Lindblad dynamics containing numerous jump operators. We demonstrate improved gate complexity in our algorithms through detailed resource estimation and validate their efficiency with numerical simulations of practical dissipative systems, which are highly consistent with exact solutions. By reducing the dependence on the number of jump operators, our algorithms address a critical bottleneck in the practical application of quantum computing to real-world problems characterized by complex dissipative dynamics.

II. MAIN RESULTS

We first list our main results and leave the proof in the next sections. We assume that operators within $\mathcal{L}$ are given as linear combinations of q Pauli strings [15],

$$H = \sum_{l=1}^q T_{0l} V_{0l}, \quad (3)$$

$$L_j = \sum_{l=1}^q T_{jl} V_{jl}, j = 1, \dots, m,$$

where $T_{jl} \geq 0$ and V_{jl} are Pauli strings. The norm depicting the intensity of $\mathcal{L}$ is defined as

$$\|\mathcal{L}\|_{\text{Pauli}} = \|H\|_{\text{Pauli}} + \sum_{j=1}^m \|L_j\|_{\text{Pauli}}^2, \quad (4)$$

where $\|H\|_{\text{Pauli}} = \sum_{l=1}^q T_{0l}$ and $\|L_j\|_{\text{Pauli}} = \sum_{l=1}^q T_{jl}$ for $j = 1, \dots, m$. Also, we use $\|\cdot\|_1$ to denote Schatten 1-norm, $\|\cdot\|_2$ to denote Euclidean norm of a vector, and $\|\cdot\|$ to denote spectral norm (i.e., operator norm) for a matrix.

Our main contributions are summarized in the following theorems. We first present a quantum algorithm for Lindbladian simulation removing the m dependence.

Theorem 1. Let $\mathcal{L}$ be a Lindbladian presented as a linear combination of q Pauli strings. For any $t > 0$ and $\epsilon > 0$, there exist quantum circuits of gate count

$$O\left(\frac{q n \tau^2}{\epsilon}\right)$$

such that sampling from these circuits implements a quantum channel $\mathcal{N}$, with $\|\mathcal{N} - e^{\mathcal{L}t}\|_{\diamond} \leq \epsilon$, where $\tau = t\|\mathcal{L}\|_{\text{Pauli}}$.

Additionally, the (t, ϵ) dependence can be improved to near optimal at the cost of introducing an $\tilde{O}(m)$ factor.

Theorem 2. Let $\mathcal{L}$ be a Lindbladian presented as a linear combination of q Pauli strings. Then, for any $t > 0$ and

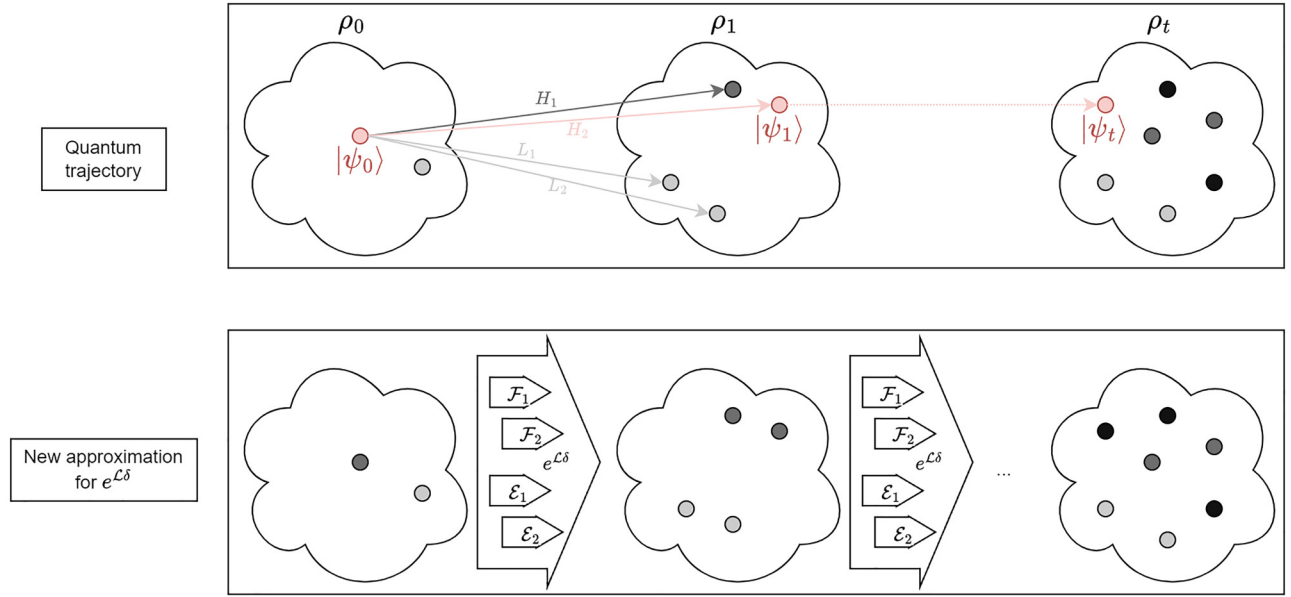


FIG. 1. In the quantum trajectory method, the state of the system is viewed as a mixture of pure states, and a trajectory is a sequence of pure states $|\psi_0\rangle, |\psi_1\rangle, \dots, |\psi_t\rangle$, the system states under a possible list of individual evolution operators. In this figure, an example Lindbladian with $H = H_1 + H_2$ and jump operators $\{L_1, L_2\}$ is presented. The mixture of all possible trajectories provides a good approximation of the evolution. Inspired by the quantum trajectory method, we introduce a new approximation for the short-time evolution of LME that is composed of efficiently implementable individual channels. In this figure, the individual channels $\mathcal{F}_1, \mathcal{F}_2$ correspond to individual Hamiltonians H_1, H_2 , and $\mathcal{E}_1, \mathcal{E}_2$ correspond to jump operators L_1, L_2 .

$\epsilon > 0$, there exists a quantum circuit of gate count

$$O\left(mq\tau \frac{\log(mq\tau/\epsilon) + n \log(\tau/\epsilon)}{\log \log(\tau/\epsilon)}\right)$$

that implements a quantum channel $\mathcal{N}$, such that $\|\mathcal{N} - e^{\mathcal{L}t}\|_{\diamond} \leq \epsilon$, where $\tau = t\|L\|_{\text{Pauli}}$.

The proof of Theorem 1 is achieved through the construction of a gate-based quantum algorithm, namely Algorithm 1. Algorithm 1 utilizes channel $\mathcal{E}$ in Eq. (8) to decompose the short-time Lindblad evolution into a mixture of individual channels representing different operators (Hamiltonian or jump operators). These channels are then implemented via sampling different gadget circuits in Figs. 2 and 3. The proof of Theorem 2 is similarly achieved through the construction of the algorithm, namely Algorithm 2. Algorithm 2 further incorporates techniques from the fractional-query method to improve the (t, ϵ) dependence, at the cost of an additional m dependence. A comparison of our results and previous results can be found in Table I.

III. NEW APPROXIMATION FOR $e^{\mathcal{L}\delta}$

Our new approximate channel design is inspired by the quantum trajectory method [30–32], an approach to simulating Markovian open quantum system numerically. Recall that in Eq. (1), the state of the system is represented

by its density matrix $\rho = \sum_j p_j |\psi_j\rangle \langle \psi_j|$, and its evolution is viewed as a deterministic transform from ρ_0 to ρ_t . The quantum trajectory method instead has another perspective. The state of the system is viewed as a mixture of pure states $\{(p_j, |\psi_j\rangle)\}_j$. Each pure state goes through a stochastic process where a random list of individual evolution operators (the Hamiltonian or a jump operator) is applied, and the state of the evolved system is the mixture of all possible result states, i.e., the average of all possible quantum trajectories.

More precisely, the following process shows how to calculate a single trajectory $|\psi(t)\rangle, t \in [0, T]$ [33] via the first-order quantum trajectory method. This allows for a full simulation since the density matrix of the evolved system can be approximated by averaging over many trajectories.

- (1) Sample the initial state $|\psi(0)\rangle$ from the initial density operator ρ_0 . Set time $t = 0$ and an adequate time step δ .

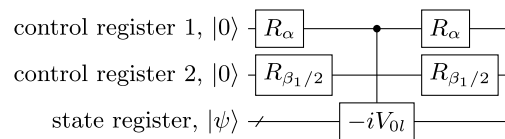


FIG. 2. Gadget circuit for simulating $\mathcal{F}_i$ modified from Ref. [36].

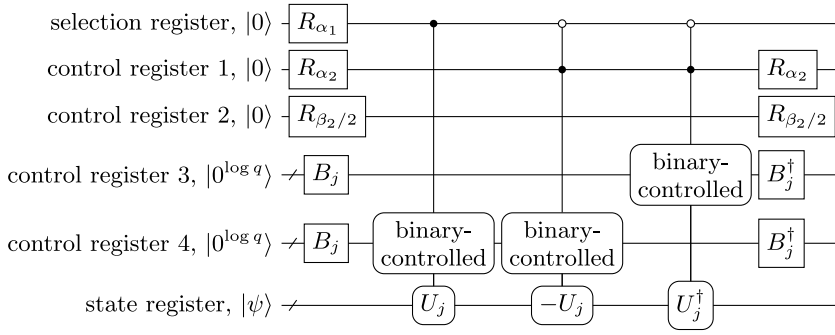


FIG. 3. Gadget circuit for simulating $\mathcal{E}_j$, where $B_j |0\rangle = \sum_{k=1}^q \sqrt{T_{jk}} |k\rangle$.

- (2) Compute the evolved state under effective Hamiltonian $|\phi_0\rangle = (I - i\delta(H - (i/2) \sum_{j=1}^m L_j^\dagger L_j)) |\psi(t)\rangle$, and its amplitude $\langle\phi_0|\phi_0\rangle = p_0$.
- (3) Compute the evolved state under a single jump operator $|\phi_j\rangle = \sqrt{\delta} L_j |\psi(t)\rangle$, and its amplitude $\langle\phi_j|\phi_j\rangle = p_j$ for $j = 1, \dots, m$.
- (4) Sample a state from the following distribution:

$$|\psi(t + \delta)\rangle = \frac{|\phi_j\rangle}{\sqrt{p_j}}, \text{ w.prob. } p_j, \text{ for } j = 0, 1, \dots, m$$

as the state of the next time step. On condition that δ is small enough, $\sum_{j=0}^m p_j \approx 1$.

- (5) Repeat until $|\psi(T)\rangle$ is obtained.

The key observation from the quantum trajectory method is to utilize *classical randomness* and avoid related computational costs by sampling.

In the following lemma, we propose a new approximation channel $\mathcal{E}$ for short-time evolution $e^{\mathcal{L}\delta}$, inspired by the quantum trajectory method. This channel $\mathcal{E}$ is the mixture of individual channels that represent the effects of Hamiltonian $(\mathcal{F}_1, \dots, \mathcal{F}_q)$ and jump operators $(\mathcal{E}_1, \dots, \mathcal{E}_m)$, respectively. To prove the lemma, we first use the fact that $I + \mathcal{L}\delta$ approximates $e^{\mathcal{L}\delta}$ up to error $(\delta \|\mathcal{L}\|_\diamond)^2$, and then we show $\mathcal{E}$ approximates $I + \mathcal{L}\delta$ up to error $(\delta \|\mathcal{L}\|_{\text{Pauli}})^2$.

Lemma 1 (quantum trajectory method via quantum channel). Let $\mathcal{L}$ be a Lindbladian presented as a linear

combination of q Pauli strings. Let $\lambda = \|\mathcal{L}\|_{\text{Pauli}}$ and $c_j = \|L_j\|_{\text{Pauli}} = \sum_{l=1}^q T_{jl}$, $j = 1, \dots, m$. Define channels

$$\mathcal{F}_l(\rho) = (I - i\lambda\delta V_{0l})\rho(I - i\lambda\delta V_{0l})^\dagger, \quad l = 1, \dots, q \quad (5)$$

and

$$\mathcal{E}_j(\rho) = A_{j0}\rho A_{j0}^\dagger + A_{j1}\rho A_{j1}^\dagger, \quad j = 1, \dots, m \quad (6)$$

where

$$\begin{aligned} A_{j0} &= I - \frac{\lambda\delta}{2c_j^2} L_j^\dagger L_j, \\ A_{j1} &= \sqrt{\frac{\lambda\delta}{c_j^2}} L_j. \end{aligned} \quad (7)$$

Let

$$\mathcal{E} = \sum_{l=1}^q \frac{T_{0l}}{\lambda} \mathcal{F}_l + \sum_{j=1}^m \frac{c_j^2}{\lambda} \mathcal{E}_j, \quad (8)$$

then

$$\|\mathcal{E} - e^{\mathcal{L}\delta}\|_\diamond \leq 5(\lambda\delta)^2. \quad (9)$$

Proof. We apply the result from [15]

$$\|(I + \delta\mathcal{L}) - e^{\mathcal{L}\delta}\|_\diamond \leq (\delta\|\mathcal{L}\|_\diamond)^2 \leq (2\lambda\delta)^2. \quad (10)$$

TABLE I. Comparison between our results and previous results on general Lindbladian simulation. For the query-based algorithm, we only consider their additional gate count. RI, repeated interaction.

Algorithm	Gate count (t, ϵ)	Gate count (m)
Algorithm 1	$O(t^2/\epsilon)$	$O(1)$
Algorithm 2	$O(t((\log(t/\epsilon))^2)/(\log \log(t/\epsilon)))$	$O(m \log m)$
Channel LCU [15]	$O(t((\log(t/\epsilon))^2)/(\log \log(t/\epsilon)))$	$O(m^2 \log m)$
Series expansion [16]	$O(t((\log(t/\epsilon))/(\log \log(t/\epsilon)))^2)$	$O(m^2 \log^2 m)$
Hamiltonian reduction (first order) [18]	$O(t^2/\epsilon)$	$O(m)$
Hamiltonian reduction (k order, $k > 1$) [18]	$O(t((t/\epsilon))^{1/k})$	$O(m^{k+1})$
RI scheme [17]	$O(t^3/\epsilon)$	$O(m^2)$

Denote the Hilbert space $\mathcal{L}$ acting on as $\mathcal{H}$. By definition,

$$\|\mathcal{E} - (I + \delta\mathcal{L})\|_{\diamond} = \max_{Q \in \mathcal{H} \otimes \mathcal{K}} \|(\mathcal{E} \otimes I_{\mathcal{K}} - (I_{\mathcal{H}} + \delta\mathcal{L}) \otimes I_{\mathcal{K}}) Q\|_1, \quad (11)$$

for an arbitrary Hilbert space $\mathcal{K}$ and $Q \in \mathcal{H} \otimes \mathcal{K}$ that $\|Q\|_1 = 1$. Bound the right-hand side and we will have a bound for $\|\mathcal{E} - (I + \delta\mathcal{L})\|_{\diamond}$. Notice that for $j = 1, \dots, m$,

$$\begin{aligned} & c_j^2(\mathcal{E}_j \otimes I_{\mathcal{K}})Q \\ &= c_j^2 Q + \lambda \delta \left((L_j \otimes I_{\mathcal{K}})Q(L_j \otimes I_{\mathcal{K}})^{\dagger} - \frac{1}{2}\{L_j^{\dagger}L_j \otimes I_{\mathcal{K}}, Q\} \right) \\ &+ \frac{\lambda^2 \delta^2}{4c_j^2} (L_j^{\dagger}L_j \otimes I_{\mathcal{K}})Q(L_j^{\dagger}L_j \otimes I_{\mathcal{K}}), \end{aligned} \quad (12)$$

and for $l = 1, \dots, q$,

$$\begin{aligned} & T_{0l}(\mathcal{F}_l \otimes I_{\mathcal{K}})Q \\ &= T_{0l}Q - i\lambda\delta[T_{0l}V_{0l} \otimes I_{\mathcal{K}}, Q] \\ &+ T_{0l}\lambda^2\delta^2(V_{0l} \otimes I_{\mathcal{K}})Q(V_{0l} \otimes I_{\mathcal{K}}). \end{aligned} \quad (13)$$

Combining with

$$\mathcal{E} \otimes I_{\mathcal{K}} = \frac{1}{\lambda} \left(\sum_{k=1}^q T_{0k}(\mathcal{F}_k \otimes I_{\mathcal{K}}) + \sum_{j=1}^m c_j^2(\mathcal{E}_j \otimes I_{\mathcal{K}}) \right), \quad (14)$$

we have

$$\begin{aligned} & (\mathcal{E} \otimes I_{\mathcal{K}} - (I_{\mathcal{H}} + \delta\mathcal{L}) \otimes I_{\mathcal{K}}) Q \\ &= \lambda\delta^2 \left(\sum_{l=1}^q T_{0l}(V_{0l} \otimes I_{\mathcal{K}})Q(V_{0l} \otimes I_{\mathcal{K}}) \right. \\ &\quad \left. + \frac{1}{4} \sum_{j=1}^m (L_j^{\dagger}L_j \otimes I_{\mathcal{K}})Q(L_j^{\dagger}L_j \otimes I_{\mathcal{K}})/c_j^2 \right), \end{aligned} \quad (15)$$

and

$$\begin{aligned} & \|\mathcal{E} \otimes I_{\mathcal{K}} - (I_{\mathcal{H}} + \delta\mathcal{L}) \otimes I_{\mathcal{K}}\|_1 \\ &\leq \lambda\delta^2 \left(c_0 + \frac{\sum_{j=1}^m \|L_j\|_1^4}{4c_j^2} \right) \leq (\lambda\delta)^2. \end{aligned} \quad (16)$$

The last line holds because $\|L_j\|_1 \leq c_j, j = 1, \dots, m$. Combined with Eq. (10) we have proved the result. ■

Notice that $\mathcal{F}_l, l = 1, \dots, q$ and $\mathcal{E}_j, j = 1, \dots, m$ are not trace preserving, but they almost do. For $j = 1, \dots, m + q$

$$\left\| \sum_k A_{jk}^{\dagger} A_{jk} - I \right\|_1 \leq (\delta\lambda)^2,$$

where $A_{m+l,0} = I - i\lambda\delta V_{0l}$ for $l = 1, \dots, q$. This suggests that in our construction each individual channel is positive and trace preserving [up to an error of $O(\delta^2)$], and thus can be implemented as individual quantum circuits.

IV. SIMULATION VIA SAMPLING

A natural way of implementing $\mathcal{E}$ is by sampling from implementations of these individual channels, which removes the dependence on the jump operator number m . Combined with the algorithmic technique of oblivious amplitude amplification for isometries (Lemma 2 in Ref. [15]), we propose Algorithm 1 for Lindbladian simulation, and analyze its approximation error and gate complexity.

A. Individual channel-simulation gadget

All quantum channels can be expressed as the partial trace of a unitary operation in a larger Hilbert space, a process known as purification. Extending this notion, we define “probabilistic purification.” It is a unitary acting on $\mathcal{H}_{\text{ctrl}} \otimes \mathcal{H}_{\text{sel}} \otimes \mathcal{H}$, and it performs corresponding quantum channel on tracing out selection space $\mathcal{H}_{\text{sel}}$ and control space $\mathcal{H}_{\text{ctrl}}$.

Definition 1. Given a channel $\mathcal{G}(\rho) = \sum_{j=1}^g A_j \rho A_j^{\dagger}$, the probabilistic purification of $\mathcal{G}$ is a unitary $U_{\mathcal{G},p}$ satisfying

$$\begin{aligned} U_{\mathcal{G},p} |0\rangle_{\text{ctrl}} |0\rangle_{\text{sel}} |\psi\rangle &= \sqrt{p} |0\rangle_{\text{ctrl}} \left(\sum_{j=1}^g |j\rangle_{\text{sel}} A_j |\psi\rangle \right) \\ &+ |\perp\rangle, \end{aligned} \quad (17)$$

where $|\perp\rangle$ is orthogonal to the subspace of the control register in the state $|0\rangle$, i.e., $\text{tr}(|0\rangle\langle 0|_{\text{ctrl}} \otimes I_{\text{sel}} \otimes I) |\perp\rangle\langle \perp| = 0$, and p is called the purification probability.

Once we can construct a probabilistic purification circuit, we can perform the corresponding quantum channel. Next we show $\{\mathcal{F}_1, \dots, \mathcal{F}_q, \mathcal{E}_1, \dots, \mathcal{E}_m\}$ have probabilistic purification circuits with probability $p = 1 - 2\lambda\delta$. These channels share a common feature: the Kraus operators are polynomials of a matrix with a Pauli string decomposition. The property motivates the following definition:

Definition 2. (n, c, d, V, q) -generated quantum channel is a quantum channel with c Kraus operators, acting on a n -qubit Hilbert space, such that

- (1) Each Kraus operator $K_j = f_j(V)$ is a polynomial of the matrix V with degree $\deg f_j < d$ for $j = 1, \dots, c$.
- (2) The matrix V is a sum of q Pauli strings, i.e., $V = \sum_{k=1}^q v_k P_k$.

The following theorem introduces the structured linear combination of unitaries (LCU) method, enabling efficient implementation of the quantum channels defined above. In contrast to the channel LCU method [15], which exhibits exponential scaling with polynomial degree d , our method achieves quadratic scaling in d .

Theorem 3 (structured LCU method). The (n, c, d, V, q) -generated quantum channel $\mathcal{C}$ can be implemented efficiently. More precisely, the probabilistic purification $U_{\mathcal{C},p}$ for $\mathcal{C}$ can be implemented with $\tilde{O}(cd^2qn)$ elementary gates, for any $p \leq p_0$ defined in Eq. (20).

Proof. For the case $c = 1$, the theorem simplifies to the addition and multiplication of block-encoded matrices. Using the standard LCU method [34,35], V has an $(n_V, \log q, 0)$ -block-encoding, where $n_V = \|V\|_{\text{Pauli}} = \sum_{k=1}^q |v_k|$, with a gate cost of $\tilde{O}(qn)$. By Ref. [34, Lemma 53], V^k has an $(n_V^k, k \log q, 0)$ -block-encoding at a cost of $\tilde{O}(kqn)$. Now consider $f_j(V) = \sum_{k=0}^{d-1} f_{jk} V^k$ for $j = 1$. Rewrite $f_j(V)$ as $\sum_{k=0}^{d-1} f_{jk} n_V^k \cdot (V^k/n_V^k)$, where (V^k/n_V^k) has a $(1, (d-1) \log q, 0)$ -block-encoding for $k = 0, \dots, d-1$. Applying [34, Lemma 54], $f_j(V)$ has an $(n_j, \log d + (d-1) \log q, 0)$ -block-encoding where $n_j = \sum_{k=0}^{d-1} |f_{jk} n_V^k|$, with a gate cost of $\tilde{O}(d^2qn)$. The circuit can be expressed as

$$U_j |0\rangle |\psi\rangle = |0\rangle \frac{f_j(V)}{n_j} |\psi\rangle + |\perp\rangle. \quad (18)$$

For the general case, we first perform a general rotation, followed by controlled U_j :

$$\begin{aligned} & |0\rangle_{\text{sel}} |0\rangle_{\text{ctrl}} |\psi\rangle \\ & \rightarrow \sum_{j=1}^c \sqrt{p_j} |j\rangle_{\text{sel}} |0\rangle_{\text{ctrl}} |\psi\rangle \\ & \rightarrow \sum_{j=1}^c \sqrt{p_j} |j\rangle_{\text{sel}} (U_j |0\rangle_{\text{ctrl}} |\psi\rangle) \\ & \rightarrow \sum_{j=1}^c |j\rangle_{\text{sel}} \left(|0\rangle_{\text{ctrl}} \sqrt{p_j} \frac{f_j(V)}{n_j} |\psi\rangle + |\perp_j\rangle \right) \\ & \rightarrow \sqrt{p_0} |0\rangle_{\text{ctrl}} \sum_{j=1}^c |j\rangle_{\text{sel}} f_j(V) |\psi\rangle + |\perp\rangle, \end{aligned} \quad (19)$$

where

$$\begin{aligned} p_0 &= \left(\sum_k n_k^2 \right)^{-1}, \\ p_j &= n_j^2 \left(\sum_k n_k^2 \right)^{-1}, j = 1, \dots, c. \end{aligned} \quad (20)$$

The total cost is $\tilde{O}(cd^2qn)$. For $p \leq p_0$, an additional control qubit with a rotation suffices. ■

We note that the same task can be achieved using channel LCU method in Ref. [15], yielding the same p_0 as in Eq. (20), but with a gate cost of $\tilde{O}(cq^d n)$.

Based on Theorem 3, we show how to implement the probabilistic purification for any channel in $\{\mathcal{F}_1, \dots, \mathcal{F}_q, \mathcal{E}_1, \dots, \mathcal{E}_m\}$. First, we consider the simulation of $\mathcal{F}_l$.

Corollary 1. Probabilistic purification of $\mathcal{F}_l$ with purification probability $p = 1 - 2\lambda\delta$ can be implemented with $O(n)$ gates for $l = 1, \dots, q$.

The channel $\mathcal{F}_l$ satisfies the condition of Theorem 3 and takes the parameters $c = 1, d = 1, q = 1$. For clarity, we still provide a full circuit construction in the proof.

Proof. Let

$$R_\alpha = \frac{1}{\sqrt{\cos \alpha + \sin \alpha}} \begin{bmatrix} \sqrt{\cos \alpha} & \sqrt{\sin \alpha} \\ \sqrt{\sin \alpha} & -\sqrt{\cos \alpha} \end{bmatrix}.$$

Circuit Fig. 2 performs the map

$$|0, 0\rangle_{\text{ctrl}} |\psi\rangle \mapsto \sqrt{q_{\alpha, \beta_1}} |0, 0\rangle_{\text{ctrl}} (I - iV_{0l} \tan \alpha) |\psi\rangle + |\perp\rangle$$

for some state $|\perp\rangle$ orthogonal to the subspace of $|0, 0\rangle_{\text{ctrl}}$, and $q_{\alpha, \beta_1} = (1/((1 + \tan \alpha)^2 (1 + \tan \beta_1)^2))$.

Suppose $\beta = 0$. Let $\alpha = \arctan \lambda\delta$, then the circuit implements a probabilistic purification of $\mathcal{F}_l$ with purification probability $p_1 = (1/((1 + \lambda\delta)^2)) > 1 - 2\lambda\delta = p$. The second control qubit and rotation gate R_{β_1} with $\beta_1 = \arctan(\sqrt{(p_1/p)} - 1)$ would tune the purification probability to p . ■

The channel $\mathcal{E}_j$ satisfies the condition of Theorem 3 if we view the term $I - L_j^\dagger L_j$ as a 2-degree polynomial of L_j . It takes the parameters $c = 2, d = 2$. Theorem 3 suggests there exists probabilistic purification $U_{\mathcal{E}_j, p}$ with gate cost $O(qn)$, while the channel LCU method costs $O(q^2n)$. In the proof below a full circuit construction is given.

Corollary 2. Probabilistic purification circuit of $\mathcal{E}_j$ with purification probability $p = 1 - 2\lambda\delta$ can be implemented with $O(qn)$ gates for $j = 1, \dots, m$.

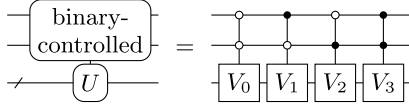


FIG. 4. Example of binary-controlled unitary matrices where the set of unitary matrices is $U = \{V_0, V_1, V_2, V_3\}$.

Proof. Define $U_j = \{V_{jl} | l = 1, \dots, q\}$, $-U_j = \{-V_{jl} | l = 1, \dots, q\}$ and $U_j^\dagger = \{V_{jl}^\dagger | l = 1, \dots, q\}$. Figure 4 depicts a set of binary-controlled unitary matrices.

Circuit Fig. 3 represents selection space with one register (selection register), and control space with four registers (control register 1, 2, 3, 4). Let us ignore control register 1 for now. The circuit maps the initial state $|0\rangle_{\text{sel}} |0\rangle_{\text{ctrl}} |\psi\rangle$ to

$$\begin{aligned} & \sqrt{p_{\alpha_1, \alpha_2}} \left[|1\rangle_{\text{sel}} |0\rangle_{\text{ctrl}} (1 + \tan \alpha_2) \sqrt{\tan \alpha_1} \frac{L_j}{c_j} |\psi\rangle \right. \\ & \quad \left. + |0\rangle_{\text{sel}} |0\rangle_{\text{ctrl}} \left(I - \tan \alpha_2 \frac{L_j^\dagger L_j}{c_j^2} \right) |\psi\rangle \right] \\ & \quad + |\perp\rangle, \end{aligned} \quad (21)$$

where

$$p_{\alpha_1, \alpha_2} = \frac{1}{(\tan \alpha_2 + 1)^2 (\tan \alpha_1 + 1)}, \quad (22)$$

and $|\perp\rangle$ is orthogonal to the subspace of $|0\rangle_{\text{ctrl}}$. By setting $\alpha_1 = \arctan((\lambda\delta/((1 + \lambda\delta/2)^2)))$ and $\alpha_2 = \arctan(\lambda\delta/2)$, circuit Fig. 3 (without control register 1) implements probabilistic purification of $\mathcal{E}_j$ with purification probability $p_2 = (1/(\lambda\delta + (1 + \lambda\delta/2)^2)) > 1 - 2\lambda\delta = p$.

Now we take control register 1 back into consideration. With $\beta_2 = \arctan(\sqrt{(p_2/p)} - 1)$, it tunes the purification probability to p . ■

By Corollaries 1 and 2 we have implemented probabilistic purification circuits $U_{\mathcal{G}, p}$ for $\mathcal{G} \in \{\mathcal{G}_1 = \mathcal{F}_1, \dots, \mathcal{G}_q = \mathcal{F}_q, \mathcal{G}_{q+1} = \mathcal{E}_1, \dots, \mathcal{G}_{q+m} = \mathcal{E}_m\}$.

B. The algorithm

Based on these gadget circuits we propose Algorithm 1 and provide a rigorous error analysis. We first consider the circuit for simulating the first segment. The indices of the sampled channel form a string $u_1 = s_{11}s_{12} \dots s_{1r} \in \{1, 2, \dots, q+m\}^r$ and corresponding Kraus operators are $\{A_{1,1}, \dots, A_{1,n_1}\}, \dots, \{A_{r,1}, \dots, A_{r,n_r}\}$. By ignoring the extra qubit in step 3a, the circuit W_1 performs the map

$$\begin{aligned} & |0\rangle_{\text{ctrl}}^{\otimes r} |0\rangle_{\text{sel}}^{\otimes r} |\psi\rangle \\ & \mapsto \sqrt{p^r} |0\rangle_{\text{ctrl}}^{\otimes r} \left(\sum_{j_1, \dots, j_r} |j_1, j_2, \dots, j_r\rangle_{\text{sel}} \prod_{k=r}^1 A_{j_k} |\psi\rangle \right) + |\perp\rangle, \end{aligned} \quad (23)$$

ALGORITHM 1. Randomized quantum algorithm for Lindbladian simulation.

Require: Lindbladian represented by Hamiltonian H and jump operators $\{L_1, \dots, L_m\}$ with Pauli string decompositions, simulation time t and precision ϵ .

1. Let $\tau = O(\lambda t)$, and $r = 2^{\lceil \log(\tau/\epsilon) \rceil}$, $\delta = t/(\tau r)$.
2. Initialize the selection registers and control registers to $|0\rangle^{\otimes r} |0\rangle^{\otimes r}$. Each selection register has 1 qubit and each control register has $2\lceil \log q \rceil + 1$ qubits.
3. For $k_1 = 1, \dots, \tau$,
 - (a) For $k_2 = 1, \dots, r$,
 - i. Sample a channel $\mathcal{G}$ from $\{\mathcal{F}_1, \dots, \mathcal{F}_q, \mathcal{E}_1, \dots, \mathcal{E}_m\}$ with probability

$$\begin{cases} \Pr(\mathcal{G} = \mathcal{F}_l) = T_{0l}/\lambda, & l = 1, \dots, q \\ \Pr(\mathcal{G} = \mathcal{E}_j) = c_j^2/\lambda, & j = 1, \dots, m \end{cases}$$

and carry out $U_{\mathcal{G}, p}$ on the k_2 th selection register, the k_2 th control register and the state register. The index of the sampled channel is saved as $s_{k_1 k_2}$.

- (b) Denote the circuit formed in step 3a in the k_1 th iteration as W_{k_1} , which acts as a constant-time simulation. Add an extra qubit as a control register and a rotation gate $R|0\rangle = p^{-r/2}/2|0\rangle + \sqrt{1 - p^{-r}/4}|1\rangle$ on it.
- (c) Use oblivious amplitude amplification for isometries on W_{k_1} , with the target state being $|0\rangle|0\rangle^{\otimes r}$ on the control registers. The result is F_{k_1} .
4. Output the result circuit $V_s = F_\tau \dots F_2 F_1$.

where $\text{tr}(|0\rangle\langle 0|_{\text{ctrl}}^{\otimes r} \otimes I_{\text{sel}}^{\otimes r} \otimes I) |\perp\rangle \langle \perp| = 0$. Notice that $p^r \geq (1 - 1/r)^r > 1/4$ by assuming $\tau \geq 2\lambda t$ and thus $R|0\rangle$ is a legal quantum state. With the extra qubit, we have

$$W_1 |\Psi\rangle = \frac{1}{2} |\Phi\rangle + \frac{\sqrt{3}}{2} |\Phi^\perp\rangle, \quad (24)$$

where

$$\begin{aligned} |\Psi\rangle &= |0\rangle |0\rangle_{\text{ctrl}}^{\otimes r} |0\rangle_{\text{sel}}^{\otimes r} |\psi\rangle, \\ |\Phi\rangle &= |0\rangle |0\rangle_{\text{ctrl}}^{\otimes r} \left(\sum_{j_1, \dots, j_r} |j_1, j_2, \dots, j_r\rangle_{\text{sel}} \prod_{k=r}^1 A_{j_k} |\psi\rangle \right). \end{aligned} \quad (25)$$

This enables us to perform oblivious amplitude amplification for isometries, which is F_1 . The following lemma restates Lemma 2 in Ref. [15] except that an index string is used to specify the sampled channels. This lemma aims to estimate the error of constant-time evolution via stacking short-time evolution and then applying the oblivious amplitude amplification. The lemma suggests that the error is of the order $O(1/r)$: the error introduced by oblivious amplitude amplification is $\|\mathcal{M} - \mathcal{E}^r\|_\diamond = O(1/r)$, and the

cumulative error from the short-time evolution channel is $\|\mathcal{E}^r - e^{\mathcal{L}r\delta}\|_\diamond \leq r\|\mathcal{E} - e^{\mathcal{L}\delta}\|_\diamond = O(1/r)$.

Lemma 2. For any state $|\psi\rangle$ and index string $u_1 = s_{11}s_{12}\dots s_{1r} \in \{1, 2, \dots, q+m\}^r$, if we define

$$F_1 = -W_1(I - 2P_1)W_1^\dagger(I - 2P_0)W_1, \quad (26)$$

where $P_0 = |0\rangle\langle 0| \otimes (|0\rangle\langle 0|)^{\otimes r}_{\text{ctrl}} \otimes I^{\otimes r}_{\text{sel}} \otimes I$ and $P_1 = |0\rangle\langle 0| \otimes (|0\rangle\langle 0|)^{\otimes r}_{\text{ctrl}} \otimes (|0\rangle\langle 0|)^{\otimes r}_{\text{sel}} \otimes I$, then

$$\|F_1|\Psi\rangle - |\Phi\rangle\|_2 \leq r(\delta\|\mathcal{L}\|_{\text{Pauli}})^2, \quad (27)$$

where $|\Psi\rangle$ and $|\Phi\rangle$ are defined in Eq. (25)

Proof. To start with, $P_0|\Phi^\perp\rangle = 0$. Then consider the circuit F_1 ,

$$\begin{aligned} \|F_1|\Psi\rangle - |\Phi\rangle\|_2 &= \|(W_1 - 4W_1P_1W_1^\dagger P_0W_1)|\Psi\rangle\|_2 \\ &= \|(I - 4P_1W_1^\dagger P_0W_1)|\Psi\rangle\|_2 \\ &= \|\Psi\rangle - 2P_1W_1^\dagger|\Phi\rangle\|_2 \\ &= \|P_1(|\Psi\rangle - 2W_1^\dagger|\Phi\rangle)\|_2. \end{aligned} \quad (28)$$

Let $|\Psi^\perp\rangle = W_1^\dagger((\sqrt{3}/2)|\Phi\rangle - \frac{1}{2}|\Phi^\perp\rangle)$, then $W_1^\dagger|\Phi\rangle = \frac{1}{2}|\Psi\rangle + (\sqrt{3}/2)|\Psi^\perp\rangle$, and

$$\|F_1|\Psi\rangle - |\Phi\rangle\|_2 = \sqrt{3}\|P_1|\Psi^\perp\rangle\|_2. \quad (29)$$

To bound $\|P_1|\Psi^\perp\rangle\|_2$ we follow the analysis in Ref. [15, Lemma 3]. Consider the operator on the state register

$$Q = \text{tr}_{\text{anc}}(|0\rangle\langle 0| \otimes (|0\rangle\langle 0|)^{\otimes r}_{\text{ctrl}} \otimes (|0\rangle\langle 0|)^{\otimes r}_{\text{sel}})W^\dagger P_0 W, \quad (30)$$

where we use tr_{anc} to denote partial trace operation that only leaves the state space. For any state $|\psi\rangle$ on the state register,

$$\begin{aligned} \langle\psi|Q|\psi\rangle &= \|P_0W|0\rangle|0\rangle^{\otimes r}_{\text{ctrl}}|0\rangle^{\otimes r}_{\text{sel}}|\psi\rangle\|_2^2 \\ &= \frac{1}{4}\langle\Psi|\Psi\rangle, \end{aligned} \quad (31)$$

while

$$\begin{aligned} |\langle\Psi|\Psi\rangle - 1| &= \left| \text{tr} \left[\left(\sum_{j_1, \dots, j_r} \prod_{l=1}^r A_{j_l}^\dagger \prod_{k=r}^1 A_{j_k} - I \right) |\psi\rangle\langle\psi| \right] \right| \\ &\leq \left\| \sum_{j_1, \dots, j_r} \prod_{l=1}^r A_{j_l}^\dagger \prod_{k=r}^1 A_{j_k} - I \right\|_1 \\ &\leq r(\delta\|\mathcal{L}\|_{\text{Pauli}})^2. \end{aligned} \quad (32)$$

This suggests all eigenvalues of Q is about $\frac{1}{4}$, i.e., $\|Q - I/4\| \leq r(\delta\|\mathcal{L}\|_{\text{Pauli}})^2/4$. Notice that

$$(Q - I/4)|\psi\rangle = \frac{\sqrt{3}}{4}(|0\rangle\langle 0|_{\text{ctrl}}^{\otimes r} |0\rangle_{\text{sel}}^{\otimes r} \otimes I)|\Psi^\perp\rangle, \quad (33)$$

then

$$\begin{aligned} \|F_1|\Psi\rangle - |\Phi\rangle\|_2 &= \sqrt{3}\|P_1|\Psi^\perp\rangle\|_2 \\ &= \sqrt{3}\|(|0\rangle\langle 0|_{\text{ctrl}}^{\otimes r} |0\rangle_{\text{sel}}^{\otimes r} \otimes I)|\Psi^\perp\rangle\|_2 \\ &= 4\|(Q - I/4)|\psi\rangle\|_2 \\ &\leq 4\|Q - I/4\| \\ &\leq r(\delta\|\mathcal{L}\|_{\text{Pauli}})^2 = O(1/r). \end{aligned} \quad (34)$$

■

Here $|\Phi\rangle$ is the purification of $\prod_{j=r}^1 \mathcal{G}_{s_{1j}}[|\psi\rangle\langle\psi|]$ for some arbitrary state $|\psi\rangle$, and $F_1|\Psi\rangle$ is an $O(1/r)$ approximation of it. This suggests F_1 simulates $\prod_{j=r}^1 \mathcal{G}_{s_{1j}}$ with error $O(1/r)$ on proper conditions, i.e.,

$$\left\| \mathcal{M}_{u_1} - \prod_{j=r}^1 \mathcal{G}_{s_{1j}} \right\|_\diamond = O(1/r), \quad (35)$$

where u_1 is defined in Lemma 2, and $\mathcal{M}_{u_1}$ is the quantum channel implemented by F_1 on tracing out selection registers and control registers, with ancillary registers initialized to $|0\rangle|0\rangle^{\otimes r}_{\text{ctrl}}|0\rangle^{\otimes r}_{\text{sel}}$. Then

$$\begin{aligned} \|\mathcal{M} - \mathcal{E}^r\|_\diamond &\leq \sum_{u_1 \in \{1, \dots, q+m\}^r} p_{u_1} \left\| \mathcal{M}_{u_1} - \prod_{j=r}^1 \mathcal{G}_{s_{1j}} \right\|_\diamond \\ &= O(1/r), \end{aligned} \quad (36)$$

where $\mathcal{M}$ is the quantum channel implemented by sampling $\mathcal{M}_{u_1}$ with probability $p_{u_1} = \prod_{j=r}^1 \Pr(\mathcal{G} = \mathcal{G}_{s_{1j}})$. If $s = u_1 \dots u_\tau = s_{11} \dots s_{r\tau}$ is sampled with probability $p_s = \prod_{j=\tau}^1 p_{u_j}$ and the selection registers and control registers are traced out, then channel $\mathcal{N} = \mathcal{M}^\tau$ is realized by Algorithm 1, and

$$\|\mathcal{N} - e^{\mathcal{L}t}\|_\diamond \leq \tau(\|\mathcal{M} - \mathcal{E}^r\|_\diamond + r\|\mathcal{E} - e^{\mathcal{L}\delta}\|_\diamond) = O(\epsilon). \quad (37)$$

By setting τ large enough, there is $\|\mathcal{N} - e^{\mathcal{L}t}\|_\diamond \leq \epsilon$.

We also analyze the total gate count in the algorithm. Figures 2 and 3 describe gadget circuits to simulate individual Hamiltonian and jump operator, respectively. The former costs $O(n)$ elementary gates, and the latter costs

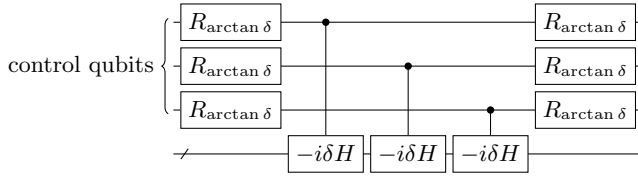


FIG. 5. Hamiltonian simulation circuit without Hamming-weight cutoff. The circuit performs $(I - i\delta H)^r$ on the state register when measuring $|0\rangle$ in all control qubits. An $r = 3$ example is presented.

$O(qn)$ elementary gates. Constant-time simulation circuit W_1 is composed of r gadget circuits and costs $O(rqn)$. Circuit $I - 2P_0$ and $I - 2P_1$ can be realized by multi-controlled Z gates on $O(r \log q)$ qubits, and they cost $O(r \log q)$ in total. Circuit F_1 costs $O(rqn)$ and V_s costs $O(\tau rqn) = O(qn\tau^2/\epsilon)$. Then we obtain Theorem 1.

At the end we review Algorithm 1. The sampling process involves two key steps: first selecting from individual channels representing the Hamiltonian and all jump operators, and then, if the Hamiltonian channel is chosen, selecting from its Pauli strings. The first step leverages classical randomness in LME to eliminate m dependence in Algorithm 1. In comparison, qDRIFT [37] focuses on sampling Pauli strings to approximate Hamiltonian dynamics efficiently, particularly for systems with many Pauli terms. Although both algorithms employ randomization, Algorithm 1 samples superoperators (channels), whereas qDRIFT samples operators (Pauli strings), highlighting a fundamental difference in their approaches to simulating quantum systems.

V. SIMULATION VIA FRACTIONAL QUERY METHOD

In this section, we apply the Hamming-weight cutoff technique to reduce (t, ϵ) dependence from $O(t^2/\epsilon)$ to $O(t((\log(t/\epsilon)^2)/(\log \log(t/\epsilon))))$, at the cost of an extra $\tilde{O}(m)$ factor in the gate complexity. In Sec. V A, we briefly review the Hamming-weight cutoff technique and its application in Hamiltonian simulation, which is generalized to Lindbladian simulation in Ref. [15]. In Sec. V B, we provide a new probabilistic purification circuit of $\mathcal{E}$ defined in Eq. (8) and shows that it meets the conditions for using the generalized Hamming-weight cutoff technique in Ref. [15]. We present the corresponding circuit design in Sec. V C. In the end, we combine them to propose Algorithm 2, and analyze its gate complexity in Sec. V D.

A. Hamming-weight cutoff

The Hamming-weight cutoff strategy is first proposed in Ref. [27] to simulate continuous-time query in discrete query model efficiently, and then applied in Hamiltonian simulation (see Fig. 6). A detailed explanation can

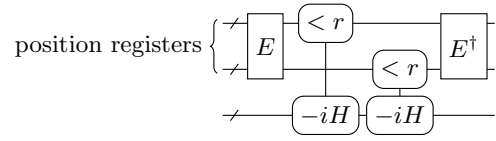


FIG. 6. Hamiltonian simulation circuit with Hamming-weight cutoff transformed from Fig. 5. The circuit approximates $(I - i\delta H)^r$ on the state register when measuring $|r\rangle^{\otimes k}$ in all control qubits. A $k = 2$ example is presented.

be found in Ref. [29, Section 2.4]. We will explain its intuition with an example of simulating e^{-iH} where H is both unitary and Hermitian. We know that $e^{-iH} = I - i\delta H + O(\delta^2)$, and $e^{-iH} = (I - i\delta H)^r + O(r\delta^2)$ where $r\delta = 1$. Therefore, with $O(r)$ call of $-iH$ we can simulate e^{-iH} with precision $O(1/r)$.

However, there is another way that approximates the original circuit and requires only $O((\log r/\log \log r))$ calls of $-iH$, and the same precision remains $O(1/r)$. We expand polynomial $(I - i\delta H)^r$ with H as the variable and only consider terms with degree $\leq k$. Equivalently, we can equip each term with an r -bit string as its index, where “0” stands for taking I and “1” stands for taking $-i\delta H$, and only consider terms whose indices’ Hamming weight is no more than k . Setting $k = O((\log \epsilon^{-1})/(\log \log \epsilon^{-1}))$ would introduce an extra error of no more than ϵ , therefore by setting $k = O((\log r/\log \log r))$ the precision remains $O(1/r)$.

To obtain a succinct circuit that approximates e^{-iH} , we need a succinct representation of control qubits and the corresponding encoded rotation operator as well. Such an encoding scheme along with a gate-efficient encoded rotation operator E is proposed in Ref. [28]. The encoding scheme $\mathcal{C}_r^k: \mathcal{H}_2^{\otimes r} \rightarrow \mathcal{H}_{r+1}^{\otimes k}$ can be interpreted as a map from the original r -qubit Hilbert space to a smaller one, and this map maintains information on the subspace expanded by computational basis states with low Hamming weights. The encoded rotation operator is a unitary operator on the smaller Hilbert space such that

$$\|\mathcal{C}_r^k[(R_{\arctan \delta} |0\rangle)^{\otimes r}] - E\mathcal{C}_r^k[|0\rangle^{\otimes r}]\|_2 \leq \epsilon_{\text{enc}} \quad (38)$$

for some encoding error ϵ_{enc} . The encoded initial state $\mathcal{C}_r^k[|0\rangle^{\otimes r}] = |r\rangle^{\otimes k}$ can be efficiently prepared.

The Hamming-weight cutoff technique is generalized to Lindbladian simulation in Ref. [15]. It turns out that, with minimal modifications, this generalized method can be applied to our method.

B. Fragment simulation gadget

Our target is to implement the probabilistic purification circuit of $\mathcal{E}$ that is suitable for the generalized Hamming-weight cutoff technique.

To start with, the probabilistic purification $U_{\mathcal{E},p}$ can be

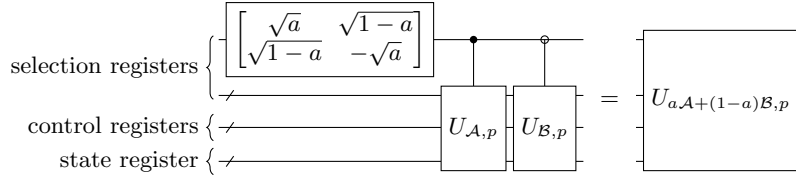


FIG. 7. Probabilistic purification circuit for the mixed channel $a\mathcal{A} + (1-a)\mathcal{B}$ for $a \in [0, 1]$, given individual channels' probabilistic purification with the same probability.

implemented efficiently by gadgets in Figs. 2 and 3, in the style presented in Figs. 7 and 8. Next, we rearrange the circuit, fitting into the four-part structure: selection preparation G , control preparation V_p , multiplexed unitary V_c , and control preparation inverse $V_p^\dagger$, which is called the structure condition. This is possible since any controlled gates commute as long as they are controlled by different binary strings on the first selection register, as shown in Fig. 9. We can also simplify these controlled gates by some simple circuit synthesis techniques (see Fig. 8). Precise construction is given in Figs. 10 and 11. The resulting circuit in Fig. 12 is a probabilistic purification circuit for $\mathcal{E}$ with probability p that is suitable for the generalized Hamming-weight cutoff technique. Next, we confirm this by the following amplitude argument. Intuitively, there is a subspace such that any state in this subspace leads to performing identity in the multiplexed unitary gate V_c . If the state after the preparation circuits V_p, G “mostly” falls into this trivial subspace, then it can be compressed, and the number of multiplexed unitary gates can be reduced. In the following, we prove that, for our construction in Fig. 12, this state indeed meets this condition. Consider the state of all the selection and control registers. The state right before the multiplexed unitary gate in Fig. 12 is

$$\sum_{l=1}^q \sqrt{\Pr(\mathcal{G} = \mathcal{F}_l)} |l, 0\rangle_{\text{sel}} |\alpha, \beta_1/2, 0, 0\rangle_{\text{ctrl}} + \sum_{j=1}^m \sqrt{\Pr(\mathcal{G} = \mathcal{E}_j)} |q+j, \alpha_1\rangle_{\text{sel}} |\alpha_2, \beta_2/2, B_j, B_j\rangle_{\text{ctrl}}, \quad (39)$$

where $|\theta\rangle = R_\theta |0\rangle$ for some angle θ , $|B_j\rangle = B_j |0\rangle$ for $j = 1, \dots, m$, and β_1, β_2 are angles shown in Figs. 2 and 3,

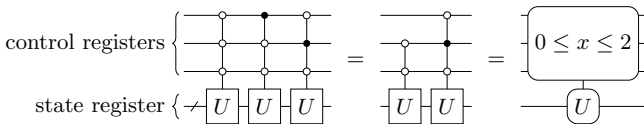


FIG. 8. Gates of the same kind but controlled on different conditions can be combined.

respectively. Consider a subspace s_I formed by a projection

$$P_I = (I \otimes |0\rangle\langle 0|)_{\text{sel}} \otimes (|0\rangle\langle 0| \otimes I \otimes I \otimes I)_{\text{ctrl}}.$$

All states in this subspace lead to performing identity I in the following multiplexed unitary gate V_c . If we measure the state in Eq. (39) by a projective measurement $\{P_I, I - P_I\}$, the probability that we obtain the outcome corresponding to P_I is

$$\begin{aligned} p_I &= \frac{\sum_{l=1}^q \Pr(\mathcal{G} = \mathcal{F}_l)}{1 + \tan \alpha} + \frac{\sum_{j=1}^m \Pr(\mathcal{G} = \mathcal{E}_j)}{(1 + \tan \alpha_1)(1 + \tan \alpha_2)} \\ &\geq \frac{1}{(1 + \tan \alpha_1)(1 + \tan \alpha_2)} \\ &\geq 1 - \frac{3}{2} \lambda \delta = 1 - \frac{3}{2r}. \end{aligned} \quad (40)$$

This is of the same scaling as in Ref. [15]. By applying the same Chernoff bound analysis we conclude that $h = O((\log r / \log \log r))$ multiplexed unitary gates do not change the order of the error, which remains $O(1/r)$.

A key technical contribution of Ref. [15] is a compression scheme suitable for circuits of the above kind. More precisely, for a probabilistic purification circuit with the same structure as Fig. 13 and satisfying the amplitude argument, there exists an encoding scheme $\mathcal{C} : (\mathcal{H}_{\text{sel}} \otimes \mathcal{H}_{\text{ctrl}})^{\otimes r} \rightarrow \mathcal{H}_{r+1}^{\otimes h} \otimes (\mathcal{H}_{\text{sel}} \otimes \mathcal{H}_{\text{ctrl}})^{\otimes h}$ and an encoded rotation operator E_{ro} such that

$$\|\mathcal{C}[(V_p(G \otimes I))|0, 0\rangle^{\otimes r}] - E_{\text{ro}}\mathcal{C}[|0, 0\rangle^{\otimes r}]\|_2 \leq \epsilon_{\text{enc}} \quad (41)$$

for some encoding error ϵ_{enc} . For different probabilistic purification circuits, only E_{ro} would be different. In Ref. [15], the construction of E_{ro} specified by the channel LCU circuit is described.

With the desired probabilistic purification circuit and efficient construction of E_{ro} specified by this circuit, we can apply the technique in Ref. [15] to achieve the same $O(\text{tpoly} \log(t/\epsilon))$ dependence.

C. Encoded rotation operator

In this subsection, we give a concrete implementation of the encoded rotation operator suitable for our purification circuit.

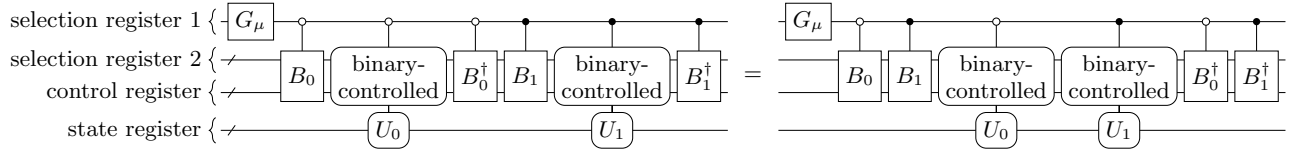


FIG. 9. Probabilistic purification circuit for the mixed channel can be rearranged to maintain the structure of binary-controlled unitary matrices.

We consider the constant-time simulation circuit W in Fig. 14. The circuit W is almost the same as W_{k_1} in step 3b of Algorithm 1, except that the sampling process is replaced with a deterministic circuit in Fig. 12. It has r gadget circuits and corresponding r selection and control registers.

Consider the computational basis of all r ancillary registers $|l_i, k_i\rangle \dots |l_r, k_r\rangle$, where l_i and k_i label the selection and control register in the i th gadget, respectively. The encoding scheme [15] compress this state to

$$|g_1, \dots, g_h\rangle |l_i, k_i\rangle \dots |l_h, k_h\rangle,$$

where $|l_i, k_i\rangle \dots |l_h, k_h\rangle$ records the first h selection and control registers that is not in s_I , and $|g_1, \dots, g_h\rangle$ encodes their position. We call $|g_1, \dots, g_h\rangle$ position register. Here $h = O((\log r / \log \log r))$ is enough according to the Hamming-weight cutoff technique. The construction of E_{ro} consists of two parts: a standard encoded rotation operator [28] on the position registers $|g_1, \dots, g_h\rangle$, followed by modified preparation circuits $V'_p (G' \otimes I)$. We only need to compose the latter part according to our probabilistic purification circuit in Fig. 12. The modified version we need is

$$V'_p (G' \otimes I) |0, 0\rangle = \frac{1}{\lambda_*} (I - P_I) V_p (G \otimes I) |0, 0\rangle \quad (42)$$

where $\lambda_* = \|(I - P_I) V_p (G \otimes I) |0, 0\rangle\|_2$.

The modified version can be done efficiently. Let $x_\theta = (1 + \tan \theta)^{-1}$ for some θ . The state we try to prepare is

$$\sum_{l=1}^q x_l |l, 0\rangle_{\text{sel}} |\xi\rangle_{\text{ctrl}} + \sum_{j=1}^m \sum_{\substack{b_1, b_2 \in \{0, 1\} \\ b_1 b_2 \neq 0}} x_{j, b_1, b_2} |q + j, b_1\rangle_{\text{sel}} |v_j\rangle_{\text{ctrl}}, \quad (43)$$

where

$$|\xi\rangle = |1, \beta_1/2, 0, 0\rangle, \quad x_l = \sqrt{\Pr(G = \mathcal{F}_l) \frac{1 - x_\alpha}{\lambda_*}}, \quad (44)$$

for $l = 1, \dots, q$, and

$$|v_j\rangle = |b_2, \beta_2/2, B_j, B_j\rangle, \quad x_{j, b_1, b_2} = \sqrt{\Pr(G = \mathcal{E}_j) \frac{(1 - x_{\alpha_1})^{b_1} (1 - x_{\alpha_2})^{b_2}}{\lambda_*}}, \quad (45)$$

for $j = 1, \dots, m, b_1 = 0, 1, b_2 = 0, 1$. Circuit G' prepares the state $\sum_{l=1}^q x_l |l, 0\rangle_{\text{sel}} + \sum_{j=1}^m \sum_{\substack{b_1, b_2 \in \{0, 1\} \\ b_1 b_2 \neq 0}} x_{j, b_1, b_2} |q + j, b_1\rangle_{\text{sel}}$, and costs at most $O(m + q)$ elementary gate. In V'_p , controlled rotation gates from V_p need modification, and the rest gates (controlled B_j gates) stay the same. The resulting circuit is shown in Fig. 15, and its gate cost remains $O(mq)$. We also need a circuit V'_c that corresponds to V_c in the original representation. V'_c has a more straightforward construction, by simply adding the condition $g_i < r$ to the original V_c .

D. The algorithm

In summation, we propose Algorithm 2, which realizes an exponential speedup in the precision parameter compared to Algorithm 1, at the cost of an extra $O(m)$ factor in gate complexity introduced by a multiplexed operation on $U_{g_j, p, j} = 1, \dots, m + q$.

For analysis of the total gate count, we consider the circuit for simulating a single segment, i.e., the circuit created from one loop of step 3. It consists of $O(1)$ encoded rotation operators E_{ro} , $O(h)$ multiplexed unitary gates V'_c , and two reflection operators. As depicted in Fig. 16, the first part of E_{ro} can be implemented with $O(h(\log r + \log \log \epsilon_2^{-1}))$ elementary gates [28], where ϵ_2 is the error introduced by an approximation of the encoding scheme and we take $\epsilon_2 = O(1/r)$. The second part consists of h copies of V'_p and G' and each costs $O(mq)$ elementary gates. Therefore, the cost of the encoded rotation operator E_{ro} is $O(h(\log r + mq))$. Multiplexed unitary gate V'_c consists of $O(mq)$ Pauli strings controlled on $O(\log r + \log(mq))$ qubits, and their cost is

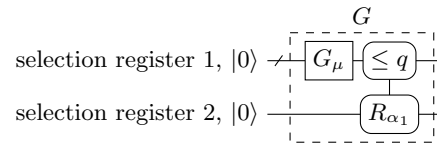
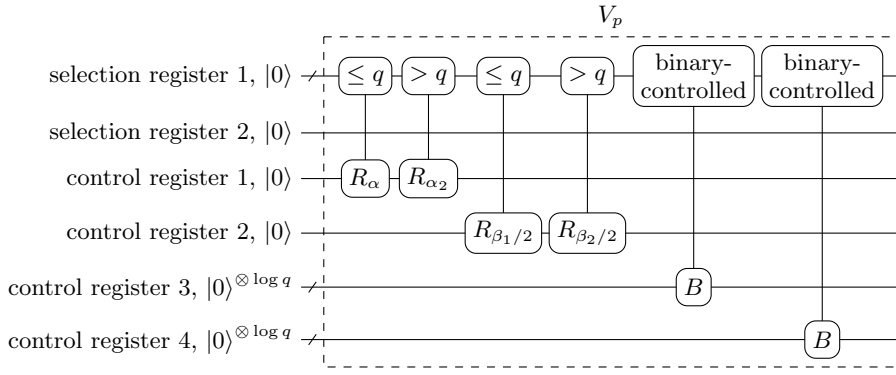


FIG. 10. Selection preparation circuit G , which prepares probability distribution.


 FIG. 11. Control preparation circuit V_p for mixing linear coefficients.

$O(mq(n + \log r + \log(mq)))$. The reflection operators can be viewed as Z gate controlled on at most $O(h(\log r + \log mq))$ qubits and their cost is $O(h(\log r + \log mq))$. Therefore, the circuit for simulating one segment has a cost of $O(h(\log r + \log mq) + mqh(n + \log r + \log(mq)) + h(\log r + \log mq)) = O(mqh(\log r + n + \log(mq)))$. There are τ segments and $h = O((\log r / \log \log r))$, $r = O(\tau/\epsilon)$, and the gate complexity of V is

$$O\left(mq\tau \frac{(\log(mq\tau/\epsilon) + n) \log(\tau/\epsilon)}{\log \log(\tau/\epsilon)}\right), \quad (46)$$

which leads to Theorem 2.

VI. EXAMPLES

The number of jump operators m in $\mathcal{L}$ of an n -qubit system can be up to 4^n , and therefore the removal of m -dependence can be an exponential improvement. We consider two examples to demonstrate such improvements. In the first example, the dissipation is given by a depolarizing channel, and the Hamiltonian has a constant Pauli norm and can be expressed as the sum of $\text{poly}(n)$ Pauli strings. The other example is the evolution of the dissipative quantum XY model [26]. In both scenarios, we

obtain better results than the previous best gate-based algorithm [15].

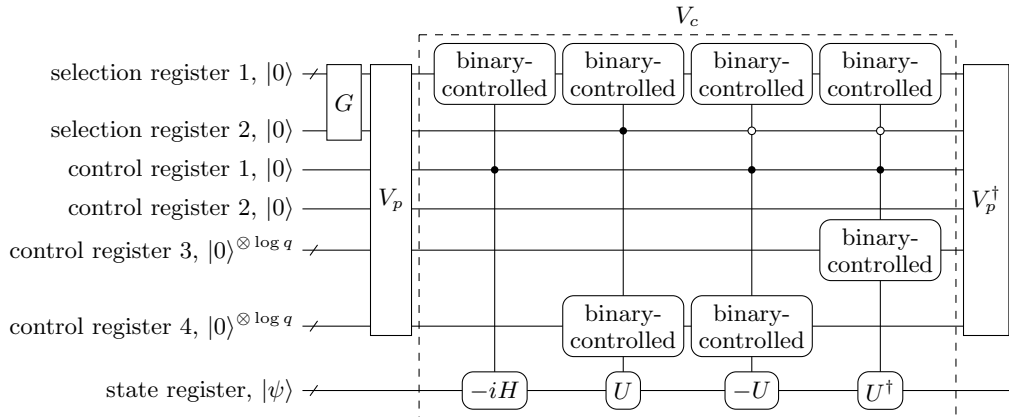
For scenario 1 listed in Table II, we first consider the simulation of the depolarizing channel. Consider the following state:

$$\rho(t) = \Delta_{\lambda(t)}(\rho(0)) = (1 - \lambda(t))\rho(0) + \lambda(t)\frac{I}{2^n}, \quad (47)$$

where $\lambda(t) = 1 - e^{-t}$. By setting different t , we are able to simulate Δ_{λ} , $\lambda \in [0, 1)$ via simulating $\rho(t)$. We assert that $\rho(t)$ can be derived from $\rho(0)$ via Lindbladian simulation. Notice that

$$\begin{aligned} \frac{d\rho(t)}{dt} &= -\rho(t) + \frac{I}{2^n} \\ &= 4^{-n} \sum_{P \in \text{Pauli}(n)} \left(P\rho(t)P^\dagger - \frac{1}{2} \{P^\dagger P, \rho(t)\} \right) \\ &= \mathcal{L}_0(\rho(t)), \end{aligned} \quad (48)$$

where $\text{Pauli}(n) = \{I, X, Y, Z\}^{\otimes n}$. This suggests $\rho(t) = e^{\mathcal{L}_0 t} \rho(0)$. Then we combine it with the given Hamiltonian


 FIG. 12. Probabilistic purification circuit for the mixed channel $\mathcal{E}$ suitable for Hamming-weight cutoff.

ALGORITHM 2. Quantum algorithm for Lindbladian simulation with exponential precision improvement

Require: Lindbladian represented by Hamiltonian H and jump operators $\{L_1, \dots, L_m\}$ with Pauli string decomposition, simulation time t and precision ϵ .

1. Let $\tau = O(\lambda t)$, $r = 2^{\lceil \log(\tau/\epsilon) \rceil}$, $\delta = t/(\tau r)$ and $h = O(\frac{\log r}{\log \log r})$.
 2. Initialize the position register, selection registers, and control registers to $|r, \dots, r\rangle (|0\rangle |0\rangle)^{\otimes h}$. The position register is composed of h subregisters and each has $\log r$ qubits. Each selection register has $\lceil \log(m+q) \rceil + 1$ qubit and each control register has $2\lceil \log q \rceil + 2$ qubits.
 3. For $k_1 = 1, \dots, \tau$,
 - (a) Carry out the encoded rotation operator E_{ro} .
 - (b) For $k_2 = 1, \dots, h$,
 - i. Carry out V'_c controlled on the k_2 th position subregister, k_2 th selection register, the k_2 th control register.
 - (c) Carry out the encoded rotation operator $E_{\text{ro}}^\dagger$.
 - (d) Reflect about the subspace where the state of the position register is $|r\rangle^{\otimes h}$ and the state of all control registers is $|0\rangle$.
 - (e) Carry out the encoded rotation operator E_{ro} .
 - (f) Carry out the inverse of the circuit created in step 3b.
 - (g) Carry out the encoded rotation operator $E_{\text{ro}}^\dagger$.
 - (h) Reflect about the subspace where the state of the position register is $|r\rangle^{\otimes h}$ and the state of all selection and control registers is $|0\rangle$.
 - (i) Carry out the circuit created from step 3b.
 - (j) Carry out the encoded rotation operator E_{ro} .
 4. Output the result circuit V .
-

H ,

$$\begin{aligned} \frac{d\rho(t)}{dt} &= -i[H, \rho] + 4^{-n} \sum_{P \in \text{Pauli}(n)} \left(P\rho(t)P^\dagger - \frac{1}{2}\{P^\dagger P, \rho(t)\} \right) \\ &= \mathcal{L}(\rho(t)), \end{aligned} \quad (49)$$

and obtain a Lindbladian $\mathcal{L}$ with a Hamiltonian and 4^n jump operators $\{(P/2^n)|P \in \text{Pauli}(n)\}$. In this scenario, the Lindbladian has the following parameters:

$$\begin{aligned} m &= 4^n, \\ q &= \text{poly}(n), q_0 = 1, \\ \|\mathcal{L}\|_{\text{Pauli}} &= O(1) + 4^n(2^{-n})^2 = O(1). \end{aligned} \quad (50)$$

where $\mathcal{L}_0$ can be described as linear combination of q_0 Pauli strings. We simulate the Lindbladian with $t = \ln((1/(1-\lambda)))$, $\epsilon = O(1)$.

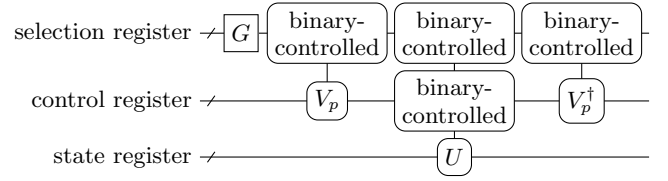


FIG. 13. Probabilistic purification circuit proposed in Ref. [15]. The Hamming-weight cutoff technique applies to circuits that both have the same structure and satisfy the amplitude condition.

In scenario 2, we consider a Lindbladian with Hamiltonian $H = -J \sum_{(i,j) \in E} (X_i X_j + Y_i Y_j)$ and n jump operators $\{Z_1, Z_2, \dots, Z_n\}$. By fully connected topology we refer to $E = \{(i,j)|1 \leq i < j \leq n\}$. We also set interaction strength to $J = -1$. The parameters of the Lindbladian are

$$\begin{aligned} m &= n, \\ q &= \frac{n(n-1)}{2}, \\ \|\mathcal{L}\|_{\text{Pauli}} &= n(n-1) + n = n^2. \end{aligned} \quad (51)$$

We simulate it with $t = n, \epsilon = O(1)$.

By substituting parameters in Theorems 1 and 2, we obtain results in Table II. While the table provides a comparison of complexity in asymptotic limit, Appendix A includes a detailed analysis with specific gate counts and runtime scaling. Appendix B supports this analysis through numerical simulations of our algorithms for Scenarios 1 and 2, demonstrating strong alignment with exact solutions.

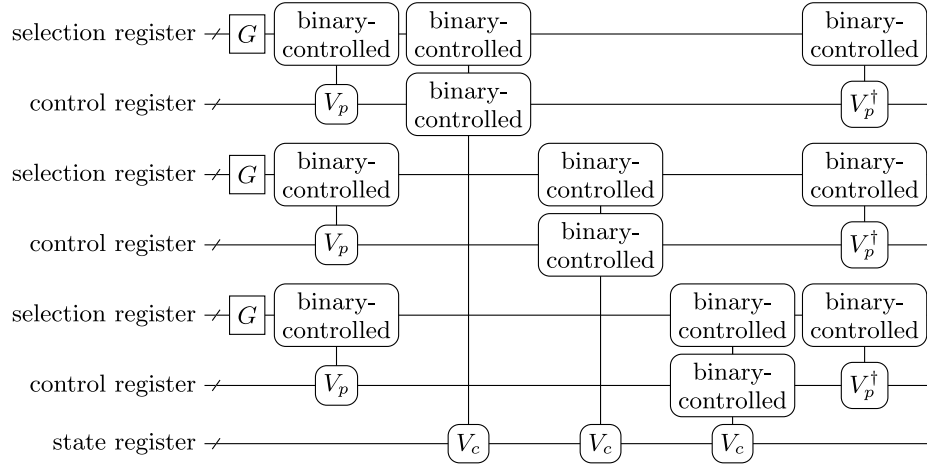
If we consider interaction strength of jump operators, denoted as $\{\gamma_1 \geq 0, \dots, \gamma_m \geq 0\}$, then the formulation of LME Eq. (1) becomes

$$\begin{aligned} \frac{d\rho(t)}{dt} &= -i[H, \rho(t)] \\ &+ \sum_{\alpha=1}^m \gamma_\alpha \left(L_\alpha \rho(t) L_\alpha^\dagger - \frac{1}{2} \{L_\alpha^\dagger L_\alpha, \rho(t)\} \right). \end{aligned} \quad (52)$$

By a substitution $L_j \rightarrow \sqrt{\gamma_j} L_j$, Theorems 1 and 2 still hold with the modified definition of $\|\mathcal{L}\|_{\text{Pauli}}$:

$$\|\mathcal{L}\|_{\text{Pauli}} = \|H\|_{\text{Pauli}} + \sum_{j=1}^m \gamma_j \|L_j\|_{\text{Pauli}}^2. \quad (53)$$

In the definition above, the contribution of interaction strength to $\|\mathcal{L}\|_{\text{Pauli}}$ is linear, which means that for a given Lindbladian, Algorithms' dependence on the interaction strength corresponds to their dependence on $\|\mathcal{L}\|_{\text{Pauli}}$ (Algorithm 1 has a quadratic dependence, while Algorithm 2 has a linear dependence, up to polylogarithmic factors).

FIG. 14. The constant-time simulation circuit W . An $r = 3$ example is presented.

VII. DISCUSSION AND CONCLUSION

In this paper, we have proposed two quantum algorithms for simulating Lindbladian evolution. Our key contribution is the introduction of a new approximation channel inspired by the quantum trajectory method, as well as the structured linear combination of unitaries method. Using the combination of the two techniques, we propose Algorithm 1, which remains efficient even with an arbitrarily large number of jump operators. Algorithm 2 further incorporates the fractional query method, achieving a near-optimal (t, ϵ) dependence at the cost of an (m, q) dependence of $\tilde{O}(mq)$. Despite this, Algorithm 2 outperforms previous algorithms that are based on linear combinations of Pauli operators [15], which have the same (t, ϵ) dependence but an (m, q) dependence of $\tilde{O}(m^2 q^2)$. Our result also improves the series truncation method [16], when the block-encoding part in Ref. [16] is implemented using the LCU method. The series truncation method has the same (t, ϵ) dependence (up to double-log factor) but an (m, q) dependence of $\tilde{O}(m^2 + mq)$. Our algorithm allows simulation with more jump operators and achieves exponentially higher precision.

The speedup on m dependence in Algorithm 2 is achieved through the integration of the new approximation channel and Theorem 3. The primary technical challenge lies in restructuring the circuits used for short-term simulation, and combining them with the Hamming-weight cutoff technique [27–29], as illustrated in Fig. 12. The encoded rotation operator's construction is intricately linked to the circuit structure for short-term simulation, preventing direct application of the method from Ref. [15]. Nevertheless, we have successfully extended the construction to attain the near-optimal (t, ϵ) dependence in Algorithm 2.

An important application of our algorithms lies in Hamiltonian eigenstate simulation, particularly ground-state preparation, enabled by recent advances in engineering Lindbladians for dissipation-driven cooling [9, 10]. For the general spin system with bulk dissipation mentioned in Ref. [10], the Lindbladian involves a large number of simple jump operators $m = O(N)$, where N is the system size, but requires a very short mixing time, $t_{\text{mix}} \sim O(\log N)$, to converge to the ground state. In this scenario, Algorithm 1 offers a substantial (potentially exponential in qubit number) speedup. Its m -independent runtime enables ground-state preparation in polylogarithmic circuit depth,

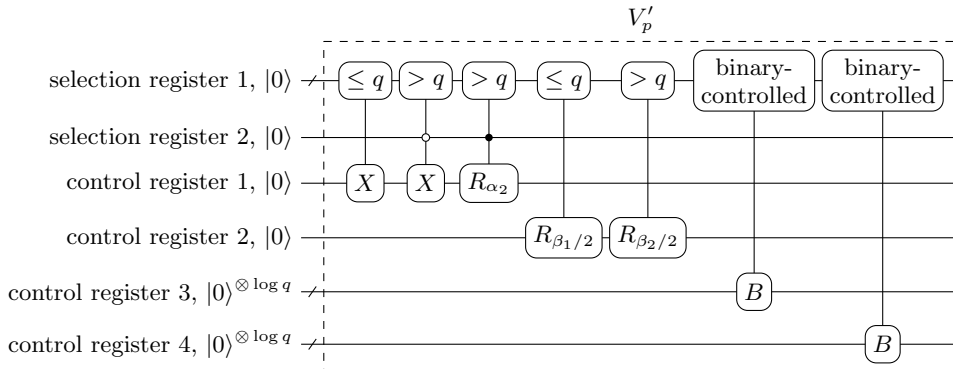
FIG. 15. Encoded control preparation circuit V'_p .

TABLE II. Comparison of the previous best gate-based algorithm [15] for Lindbladian simulation and our algorithms. In the first scenario, we consider “depolarized” Hamiltonian simulation on n qubits, with constant precision ϵ and constant depolarizing rate λ . In the second scenario, we consider the simulation of the n -qubit dissipative quantum XY model (as defined in Ref. [26]) with full connection topology, interaction strength $J = -1$, time $t = n$, and constant precision. Notice that for scenario 1, we use a strengthened version of Theorem 1, where the gate count can be reduced to $O((q_0 n \tau^2 / \epsilon))$, where q_0 is the maximum Pauli string number for jump operators only.

Algorithm	Scenario 1	Scenario 2
Algorithm 1	$O(n)$	$O(n^7)$
Algorithm 2	$O(\text{poly}(n)2^{2n})$	$O(n^6 \log n)$
Channel LCU scheme [15]	$O(\text{poly}(n)2^{4n})$	$O(n^9 \log n)$

demonstrating this specific problem resides not only in QMA but also in QNC. In contrast, previous algorithms with a polynomial dependence on m would require $\Omega(N)$ -depth circuits. Our results do not contradict the QMA hardness of general local Hamiltonian problems: “hard” Hamiltonians with exponentially small gaps [38] exhibit exponentially long mixing times, where the no-fast-forwarding theorem [39,40] forbids subexponential simulation costs for any Lindbladian-based method. Furthermore, the cost of our algorithms depends on both the initial fidelity with the target state, $F_0 = F(\rho_0, \sigma)$, and the required final infidelity η . Intuitively, a lower F_0 necessitates longer evolution to amplify the target state component. To achieve a target fidelity of $F(\rho, \sigma) \geq 1 - \eta$, the gate complexity of Algorithm 1 scales as $O((\log((1 - F_0)/\eta))^2 \eta^{-1})$ while that of Algorithm 2 scales as $O(\log((1 - F_0)/\eta) \log(\eta^{-1}))$, ignoring sublogarithmic factors. The logarithmic dependence on $1 - F_0$ in both algorithms originates from the mixing time property: each t_{mix} evolution period halves the trace distance to the target state.

Several open questions remain. It would be interesting to explore the possibility of combining the merits of Algorithms 1 and 2. Specifically, we wonder if it is possible to develop a quantum algorithm for Lindbladian simulation that achieves an ϵ dependence of $O(\log \epsilon^{-1})$ while being

independent of m concerning gate complexity. An intriguing variant to consider is the trade-off between the fractional query approach and the sampling approach, which benefits in cases with moderate values of m . Additionally, establishing a lower bound on the (m, q, t, ϵ) dependence for Lindbladian simulation or the (q, t, ϵ) dependence for Hamiltonian simulation would be of significant interest.

An extension of the quantum trajectory method has recently been proposed [41], it would be interesting to further combine the extended method with the structured LCU method proposed in this paper, and investigate the performance of the resulting quantum algorithm.

We also look forward to applications of structured LCU method (Theorem 3) in other tasks related to the digital implementation of quantum channels.

ACKNOWLEDGMENTS

We gratefully acknowledge Benjamin Desef, the author of the `yquant` LaTeX package, for providing instructions on drawing specific complex quantum gates. We thank Chunhao Wang, Wenjun Yu for insightful discussions. This work was supported in part by the National Natural Science Foundation of China Grants No. 62325210, No. 92465202, and No. 12204489. Q.Z. acknowledges funding from Innovation Program for Quantum Science and Technology via Project No. 2024ZD0301900, National Natural Science Foundation of China (NSFC) via Projects No. 12347104 and No. 12305030, Guangdong Basic and Applied Basic Research Foundation via Project No. 2023A1515012185, Hong Kong Research Grant Council (RGC) via No. 27300823, No. N_HKU718/23, and No. R6010-23, Guangdong Provincial Quantum Science Strategic Initiative No. GDZX2303007, HKU Seed Fund for Basic Research for New Staff via Project No. 2201100596.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available. The data are available from the authors upon reasonable request.

APPENDIX A: RESOURCE ESTIMATION

In this section, we provide resource estimation for various Lindbladian simulation algorithms including Algorithms 1, 2, and the algorithm in Ref. [15]. We analyze the constants behind the big- O notation for the algorithms and illustrate in Figs. 17 and 18 how their gate counts vary with time on the target system mentioned in Table II.

In the following analysis, we use the following results:

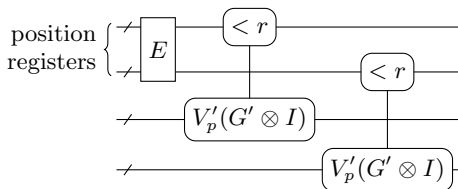


FIG. 16. The encoded rotation operator E_{ro} . It consists of two parts: a standard encoded rotation operator E [28] on the position registers $|g_1, \dots, g_n\rangle$, followed by modified preparation circuits $V'_p(G' \otimes I)$. An $h = 2$ example is shown.

- (a) Multicontrolled elementary gate (or extra control bit) costs at most $2n - 1$ elementary gates where n is the number of control qubits. Nielsen and Chuang [42], Nie *et al.* [43].
- (b) Real state preparation costs at most $2^n(2n - 4) + 4$ elementary gates, i.e., at most $2q \log q$ for B_j and $2(m + q) \log(m + q)$ for G_μ . Grover and Rudolph [44], Sun *et al.* [45].
- (c) n -qubit quantum adder is available in $n \log n$ elementary gates. Draper [46].

For Algorithm 1, we need to specify the exact value of τ to know the constant. We accompany the result of Lemma 2 with more details. Let $|\Psi'\rangle = F|\Psi\rangle$, then

$$\begin{aligned}
 & \left\| \mathcal{M}_{u_1} - \prod_{j=r}^1 \mathcal{G}_{s_{1j}} \right\|_1 \\
 &= \max_{|\psi\rangle} \left\| \mathcal{M}_{u_1}[|\psi\rangle\langle\psi|] - \prod_{j=r}^1 \mathcal{G}_{s_{1j}}[|\psi\rangle\langle\psi|] \right\|_1 \\
 &\leq \| |\Psi'\rangle\langle\Psi'| - |\Phi\rangle\langle\Phi| \|_1 \\
 &= 2\sqrt{1 - |\langle\Psi'|\Phi\rangle|^2} \\
 &\leq 2 \| |\Psi'\rangle - |\Phi\rangle \|_2 \leq 2r(\delta \|\mathcal{L}\|_{\text{Pauli}})^2. \quad (\text{A1})
 \end{aligned}$$

For the second line, notice that partial trace is a CPTP map (which maps $|\Psi'\rangle$ to $\mathcal{M}_{u_1}[|\psi\rangle\langle\psi|]$ and $|\Phi\rangle$ to $\prod_{j=r}^1 \mathcal{G}_{s_{1j}}[|\psi\rangle\langle\psi|]$) and thus is contractive. For the third line, we use the fact that the trace distance between the pure states is related to their fidelity, e.g., see Ref. [47, Theorem

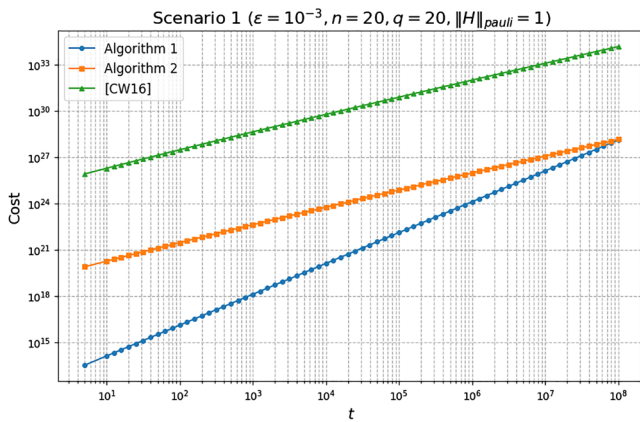


FIG. 17. Gate count estimation for Algorithms 1, 2 and the algorithm in Ref. [15] in Scenario 1 varying with time t . We set the same precision as in the comparison in Ref. [37].

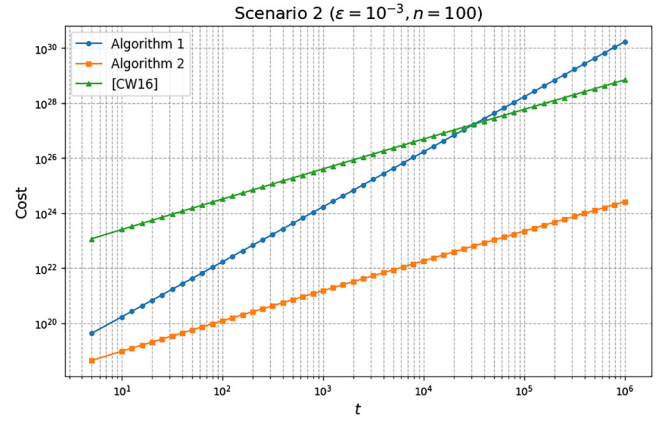


FIG. 18. Gate count estimation for Algorithms 1, 2 and the algorithm in Ref. [15] in Scenario 2 varying with time t . We set the same precision as in the comparison in Ref. [37].

9.3.1]. Then

$$\begin{aligned}
 \|\mathcal{M} - \mathcal{E}^r\|_1 &\leq 2r(\delta \|\mathcal{L}\|_{\text{Pauli}})^2, \\
 \|\mathcal{N} - e^{\mathcal{L}^t}\|_1 &\leq \tau(\|\mathcal{M} - \mathcal{E}^r\|_1 + r\|\mathcal{E} - e^{\mathcal{L}^\delta}\|_1) \\
 &\leq \tau(2r(\delta \|\mathcal{L}\|_{\text{Pauli}})^2 + 5r(\delta \|\mathcal{L}\|_{\text{Pauli}})^2) \\
 &\leq 7\tau r(\delta \|\mathcal{L}\|_{\text{Pauli}})^2. \quad (\text{A2})
 \end{aligned}$$

Let $\tau = \sqrt{7}t\|\mathcal{L}\|_{\text{Pauli}}$ in Algorithm 1 and the error is bounded to ϵ . The cost of gadget Fig. 2 is $c_{\text{Pauli}} \leq n + 4$ and the cost of gadget Fig. 3 is $c_{\text{jump}} \leq 14(q \log q + qn)$. The total cost of Algorithm 1 is at most

$$\begin{aligned}
 & \tau(3r \max\{c_{\text{Pauli}}, c_{\text{jump}}\} + 2(2r - 1)) \\
 &\leq 300 \cdot \frac{(t\|\mathcal{L}\|_{\text{Pauli}})^2}{\epsilon} \cdot q(\log q + n). \quad (\text{A3})
 \end{aligned}$$

For Algorithm 2, we need the exact values of h and τ . Encoded selection preparation circuit G costs at most $2(m + q) \log(m + q)$. Encoded control preparation circuit V'_p costs $2(m + q) \log(m + q) + 2((2q + 1) \log(m + q)) = (2m + 6q + 1) \log(m + q)$. Finally, V'_c costs at most $6mq(\log(m + q) + n) \leq 6mqn \log(m + q)$. In total, E_{ro} costs at most $7hmqn \log(m + q) \log r$ plus the cost of the original encoding operator E . The cost of the original encoding operator E (as described in Ref. [28, Section 4.4]) is at most $9h \log r$. Therefore E_{ro} costs at most $8hmqn \log(m + q) \log r$ for $mqn \geq 9$. The total cost of Algorithm 2 is at most

$$\begin{aligned}
 & \tau(5\#E_{\text{ro}} + 3 \cdot 6hmqn \log(m + q) + 2 \cdot 2h \log r) \\
 &\leq 60hmqn\tau \log(m + q) \log r, \quad (\text{A4})
 \end{aligned}$$

where $\#E_{\text{ro}}$ refers to the cost of the encoded rotation operator E_{ro} .

We estimate the value of τ and h from the following argument: the Hamming weight cutoff technique can be intuitively explained as using $\mathcal{C}'' = \sum_{k=0}^h \binom{r}{k} (\mathcal{L}\delta)^k$ to approximate $\mathcal{C}' = \sum_{k=0}^r \binom{r}{k} (\mathcal{L}\delta)^k = (I + \mathcal{L}\delta)^r$, which is an approximation for $\mathcal{C} = e^{\mathcal{L}\delta}$ itself. We choose r (thus τ) and h such that

$$\|\mathcal{C}'' - \mathcal{C}\|_{\diamond} \leq \epsilon/\tau. \quad (\text{A5})$$

From Ref. [15, Appendix B], $\|\mathcal{C}' - \mathcal{C}\|_{\diamond} \leq r(\delta \|\mathcal{L}\|_{\diamond})^2$. Combined with

$$\begin{aligned} \|\mathcal{C}'' - \mathcal{C}'\|_{\diamond} &\leq \sum_{k=h+1}^r \binom{r}{k} (\delta \|\mathcal{L}\|_{\diamond})^k \\ &\leq \binom{r}{h+1} (\delta \|\mathcal{L}\|_{\diamond})^{h+1} \\ &\leq \frac{(r\delta \|\mathcal{L}\|_{\diamond})^{h+1}}{(h+1)!}, \end{aligned} \quad (\text{A6})$$

then

$$\|\mathcal{C}'' - \mathcal{C}\|_{\diamond} \leq r(\delta \|\mathcal{L}\|_{\diamond})^2 + (r\delta \|\mathcal{L}\|_{\diamond})^{h+1}/(h+1)!. \quad (\text{A7})$$

Let $r\delta \|\mathcal{L}\|_{\diamond} = c$ be a constant, then

$$\|\mathcal{C}'' - \mathcal{C}\|_{\diamond} \leq c^2\epsilon/\tau + c^{h+1}/(h+1)!. \quad (\text{A8})$$

Let $c = 1/2$ and $h+1 = \log(\tau/\epsilon)$, the right-hand side is bounded below ϵ/τ . Then $\tau = (t\|\mathcal{L}\|_{\diamond}/c) = 2t\|\mathcal{L}\|_{\diamond}$, and $h = \log(t\|\mathcal{L}\|_{\diamond}/\epsilon)$. The total cost is at most

$$\begin{aligned} &60hmqn\tau \log(m+q) \log r \\ &\leq 120(\log(t\|\mathcal{L}\|_{\diamond}/\epsilon) + 1)^2 mqn(t\|\mathcal{L}\|_{\diamond}) \log(m+q) \\ &\leq 240(\log(t\|\mathcal{L}\|_{\text{Pauli}}/\epsilon) + 2)^2 mqn(t\|\mathcal{L}\|_{\text{Pauli}}) \\ &\log(m+q). \end{aligned} \quad (\text{A9})$$

We estimate the cost of Ref. [15] by replacing the circuit in Fig. 12 with the channel LCU method in Ref. [15]. The cost is estimated to be

$$\begin{aligned} &\tau(5\#E'_{\text{ro}} + 3 \cdot 2hm^2q^2n \log m + 2 \cdot 2h \log r) \\ &\leq 6hm^2q^2n\tau \log m \log r, \end{aligned} \quad (\text{A10})$$

where $\#E'_{\text{ro}}$ refers to the cost of encoding operator in Ref. [15]. In Ref. [15], controlled- B gate costs $2mq^2 \log m$ and G gate costs $2 \log m$, therefore $\#E'_{\text{ro}} = 9h \log r + 2mq^2n \log m$. By the same argument for τ and h , we

estimate the cost of the algorithm in Ref. [15] to be at most

$$\begin{aligned} &6hm^2q^2n\tau \log m \log r \\ &\leq 12(\log(t\|\mathcal{L}\|_{\diamond}/\epsilon) + 1)^2 m^2q^2n(t\|\mathcal{L}\|_{\diamond}) \log m \\ &\leq 24(\log(t\|\mathcal{L}\|_{\text{Pauli}}/\epsilon) + 2)^2 m^2q^2n(t\|\mathcal{L}\|_{\text{Pauli}}) \log m. \end{aligned} \quad (\text{A11})$$

Now we can illustrate how gate counts of different Lindbladian simulation algorithms vary with time t . The systems are chosen to be dissipative quantum XY model defined in Eq. (51) and “depolarized” Hamiltonian simulation defined in Eq. (50).

APPENDIX B: NUMERICAL SIMULATION

In this section, we present the results of numerical simulations for Algorithms 1 and 2 under two different scenarios and compare these with the exact solutions.

For Scenario 1, we analyze the Hamiltonian of the one-dimensional transverse-field Ising model. The Hamiltonian and jump operators are defined as follows:

$$\begin{aligned} H &= -J \sum_{i=0}^{n-2} Z_i Z_{i+1} - h \sum_{i=0}^{n-1} X_i, \\ L_P &= P \in \{I, X, Y, Z\}^{\otimes n}, \end{aligned} \quad (\text{B1})$$

where $J = -1$ and $h = 0.5$. The initial state is $|\uparrow\uparrow\uparrow\uparrow\rangle = |0000\rangle$. For both algorithms, we set the precision $\epsilon = 10^{-2}$ and the time span is $[0, 2]$.

For Algorithm 1, we conduct the simulation with 4×10^2 samples. The mixture of these samples is presented as the final result. The parameters are

$$\begin{aligned} \tau &= 30 \leq \sqrt{7}t\|\mathcal{L}\|_{\text{Pauli}}, \\ r &= 3 \times 10^3 = \tau/\epsilon. \end{aligned} \quad (\text{B2})$$

For Algorithm 2, the parameters are

$$\begin{aligned} \tau &= 48 \leq 2t\|\mathcal{L}\|_{\diamond}, \\ r &= 4.8 \times 10^3 = \tau/\epsilon, \\ h &= 10 \leq \log(\tau/\epsilon) - 1. \end{aligned} \quad (\text{B3})$$

In Scenario 2, the dynamic system is the dissipative quantum XY model with a 2×2 grid topology. The Hamiltonian strength is $J = -1$ and jump operator strength $\gamma = 0.1$. The Hamiltonian and jump operators are

$$\begin{aligned} H &= \sum_{(i,j) \in E} (X_i X_j + Y_i Y_j), \\ E &= \{(0, 1), (0, 2), (1, 3), (2, 3)\}, \\ L_j &= \sqrt{\gamma} Z_j, \quad \gamma = 0.1, \quad j = 0, 1, 2, 3. \end{aligned} \quad (\text{B4})$$

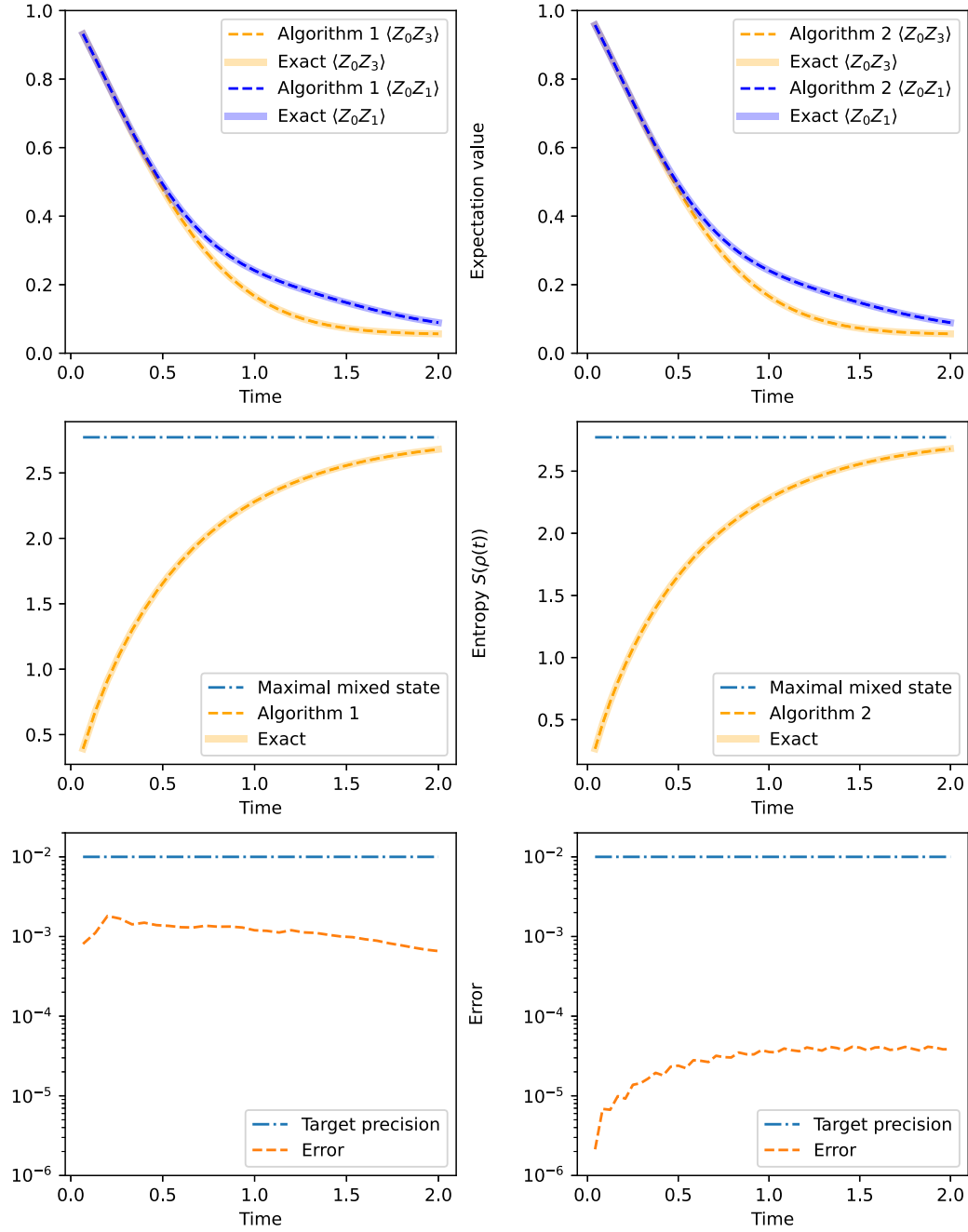


FIG. 19. Simulation results for the depolarized transverse-field Ising model by Algorithm 1 (left) and Algorithm 2 (right). The first row depicts the evolution of correlation functions $\langle Z_0 Z_1 \rangle$ and $\langle Z_0 Z_3 \rangle$. The second row depicts the evolution of von Neumann entropy. The third row depicts the error $\|\rho_t - \tilde{\rho}_t\|_1$ between exact solution ρ_t and our result $\tilde{\rho}_t$.

The initial state is given by

$$\bigotimes_{j=0}^3 (\cos \theta_j |0\rangle + \sin \theta_j |1\rangle), \quad (\text{B5})$$

with $\{\theta_0, \dots, \theta_3\}$ sampled uniformly from $[0, 2\pi]$. The precision is set to $\epsilon = 10^{-2}$ and the time span is $[0, 15]$.

For Algorithm 1, we simulate using 4×10^2 samples. The parameters are

$$\begin{aligned} \tau &= 3 \times 10^2 \leq \sqrt{7}t \|\mathcal{L}\|_{\text{Pauli}}, \\ r &= 3 \times 10^4 = \tau/\epsilon. \end{aligned} \quad (\text{B6})$$

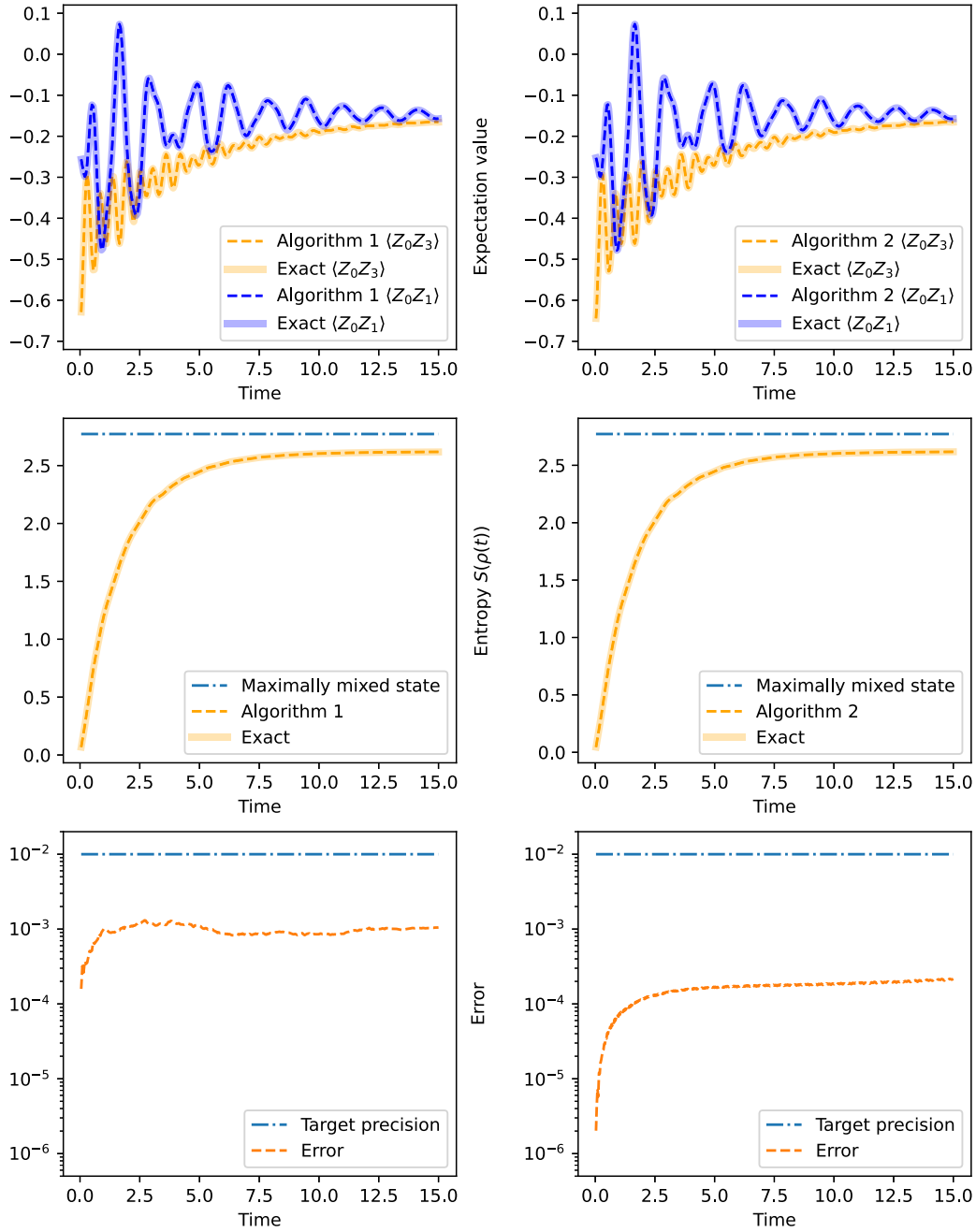


FIG. 20. Simulation results for the dissipative quantum XY model [26] by Algorithm 1 (left) and Algorithm 2 (right). The first row depicts the evolution of correlation functions $\langle Z_0 Z_1 \rangle$ and $\langle Z_0 Z_3 \rangle$. The second row depicts the evolution of von Neumann entropy. The third row depicts the error $\|\rho_t - \tilde{\rho}_t\|_1$ between exact solution ρ_t and our result $\tilde{\rho}_t$.

For Algorithm 2, the parameters are:

$$\begin{aligned} \tau &= 4.8 \times 10^2 \leq 2t \|\mathcal{L}\|_{\diamond}, \\ r &= 4.8 \times 10^4 = \tau/\epsilon, \\ h &= 14 \leq \log(\tau/\epsilon) - 1. \end{aligned} \quad (\text{B7})$$

These settings are aligned with the resource estimation provided in Appendix A and aim to achieve a target precision of ϵ . The simulation produces τ data points, derived

from the simulated states $\tilde{\rho}_1, \dots, \tilde{\rho}_{\tau-1}, \tilde{\rho}_{\tau}$ at time points $(1/\tau)t, \dots, ((\tau-1)/\tau)t, t$.

For comparison, the exact solution $\rho_1, \dots, \rho_{\tau}$ is computed using `scipy.integrate.solve_ivp`, with `rtol=1e-5` and `atol=1e-6`.

The first two rows of Figs. 19 and 20 compare our simulation results with the exact solution, confirming the correctness of both algorithms. The third row presents the simulation error. For both algorithms, the error remains

within the target precision (indicated by the horizontal line), consistent with expectations based on Appendix A. However, several noteworthy patterns emerge, which we briefly analyze below:

- (a) **Fluctuations in simulation errors for Algorithm 1:** The simulation error arises from two sources: the bias $\|\mathbb{E}[\tilde{\rho}_j] - \rho_j\|_1$ and the variation $\mathbb{E}[\|\mathbb{E}[\tilde{\rho}_j] - \tilde{\rho}_j\|_1]$, where $\tilde{\rho}_j$ is the simulated state and ρ_j the exact state. The variation explains the fluctuation.
- (b) **Higher-than-expected precision:** The actual simulation error is significantly smaller than the target precision, indicating that the resource estimation may be overly conservative. This suggests that our parameter setting in Appendix A can potentially yield higher precision than theoretically guaranteed.
- (c) **Precision differences between the two algorithms:** Given the same target precision, Algorithm 2 achieves a notably higher precision compared to Algorithm 1. This indicates that the resource bounds for Algorithm 2 may be further tightened. Additionally, the variation error in Algorithm 1 may partially explain the observed difference.

-
- [1] R. P. Feynman, Simulating physics with computers, *Int. J. Theor. Phys.* **21**, 467 (1982).
 - [2] A. Nitzan, *Chemical Dynamics in Condensed Phases: Relaxation, Transfer and Reactions in Condensed Molecular Systems* (Oxford University Press, Oxford, 2006).
 - [3] V. May and O. Kühn, *Charge and Energy Transfer Dynamics in Molecular Systems* (John Wiley & Sons, Ltd, Weinheim, Germany, 2011).
 - [4] U. Weiss, *Quantum Dissipative Systems* (World Scientific, Singapore, 2012), 4th ed. <https://www.worldscientific.com/doi/pdf/10.1142/8334>.
 - [5] P. M. Harrington, E. J. Mueller, and K. W. Murch, Engineered dissipation for quantum information science, *Nat. Rev. Phys.* **4**, 660 (2022).
 - [6] S. P. Wolski, D. Lachance-Quirion, Y. Tabuchi, S. Kono, A. Noguchi, K. Usami, and Y. Nakamura, Dissipation-based quantum sensing of magnons with a superconducting qubit, *Phys. Rev. Lett.* **125**, 117701 (2020).
 - [7] M. Raghunandan, J. Wrachtrup, and H. Weimer, High-density quantum sensing with dissipative first order transitions, *Phys. Rev. Lett.* **120**, 150501 (2018).
 - [8] Z. Chen, Y. Lu, H. Wang, Y. Liu, and T. Li, Quantum Langevin dynamics for optimization, *Commun. Math. Phys.* **406**, (2025).
 - [9] Z. Ding, C.-F. Chen, and L. Lin, Single-ancilla ground state preparation via Lindbladians, *Phys. Rev. Res.* **6**, 033147 (2024).
 - [10] Y. Zhan, Z. Ding, J. Huhn, J. Gray, J. Preskill, G. K.-L. Chan, and L. Lin, Rapid quantum ground state preparation via dissipative dynamics, [arXiv:2503.15827](https://arxiv.org/abs/2503.15827).
 - [11] H.-B. Chen, C. Gneiting, P.-Y. Lo, Y.-N. Chen, and F. Nori, Simulating open quantum systems with Hamiltonian ensembles and the nonclassicality of the dynamics, *Phys. Rev. Lett.* **120**, 030403 (2018).
 - [12] H. Kamakari, S.-N. Sun, M. Motta, and A. J. Minnich, Digital quantum simulation of open quantum systems using quantum imaginary-time evolution, *PRX Quantum* **3**, 010320 (2022).
 - [13] 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).
 - [14] C. W. Kim, J. M. Nichol, A. N. Jordan, and I. Franco, Analog quantum simulation of the dynamics of open quantum systems with quantum dots and microelectronic circuits, *PRX Quantum* **3**, 040308 (2022).
 - [15] R. Cleve and C. Wang, in *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, Leibniz International Proceedings in Informatics (LIPIcs), Vol. 80, edited by I. Chatzigiannakis, P. Indyk, F. Kuhn, and A. Muscholl (Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2017), pp. 17:1–17:14.
 - [16] X. Li and C. Wang, in *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, Leibniz International Proceedings in Informatics (LIPIcs), Vol. 261, edited by K. Etessami, U. Feige, and G. Puppis (Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2023) pp. 87:1–87:20.
 - [17] M. Pocrnic, D. Segal, and N. Wiebe, Quantum simulation of Lindbladian dynamics via repeated interactions, *J. Phys. A: Math. Theor.* **58**, 305302 (2025).
 - [18] Z. Ding, X. Li, and L. Lin, Simulating open quantum systems using Hamiltonian simulations, *PRX Quantum* **5**, 020332 (2024).
 - [19] J. Kato, K. Wada, K. Ito, and N. Yamamoto, Exponentially accurate open quantum simulation via randomized dissipation with minimal ancilla, [arXiv:2412.19453](https://arxiv.org/abs/2412.19453).
 - [20] D. Patel and M. M. Wilde, Wave matrix Lindbladization I: Quantum programs for simulating Markovian dynamics, *Open Syst. Inf. Dyn.* **30**, 2350010 (2023).
 - [21] D. Patel and M. M. Wilde, Wave matrix Lindbladization II: General Lindbladians, linear combinations, and polynomials, *Open Syst. Inf. Dyn.* **30**, 2350014 (2023).
 - [22] 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).
 - [23] S.-J. Wei, D. Ruan, and G.-L. Long, Duality quantum algorithm efficiently simulates open quantum systems, *Sci. Rep.* **6**, 30727 (2016).
 - [24] S. Wei, T. Wang, D. Ruan, and G. Long, Advances in quantum computing (in Chinese), *Sci. Sin. Inform.* **47**, 1277 (2017).
 - [25] E. Borrás and M. Marvian, Quantum algorithm to simulate Lindblad master equations, *Phys. Rev. Res.* **7**, 023076 (2025).
 - [26] 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, *Quantum* **9**, 1765 (2025).

- [27] R. Cleve, D. Gottesman, M. Mosca, R. D. Somma, and D. Yonge-Mallo, in *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09 (Association for Computing Machinery, New York, NY, USA, 2009), pp. 409–416.
- [28] D. W. Berry, R. Cleve, and S. Gharibian, Gate-efficient discrete simulations of continuous-time quantum query algorithms, *Quantum Inf. Comput.* **14**, 1 (2014).
- [29] R. Kothari, *Efficient algorithms in quantum query complexity*, PhD thesis, University of Waterloo, 2014.
- [30] J. Dalibard, Y. Castin, and K. Mølmer, Wave-function approach to dissipative processes in quantum optics, *Phys. Rev. Lett.* **68**, 580 (1992).
- [31] K. Mølmer, Y. Castin, and J. Dalibard, Monte Carlo wave-function method in quantum optics, *J. Opt. Soc. Am. B* **10**, 524 (1993).
- [32] R. Dum, P. Zoller, and H. Ritsch, Monte Carlo simulation of the atomic master equation for spontaneous emission, *Phys. Rev. A* **45**, 4879 (1992).
- [33] A. J. Daley, Quantum trajectories and open many-body quantum systems, *Adv. Phys.* **63**, 77 (2014).
- [34] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019 (Association for Computing Machinery, New York, NY, USA, 2019), pp. 193–204.
- [35] A. M. Childs and N. Wiebe, Hamiltonian simulation using linear combinations of unitary operations, *Quantum Inf. Comput.* **12**, 901 (2012).
- [36] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '14 (Association for Computing Machinery, New York, NY, USA, 2014), pp. 283–292.
- [37] E. Campbell, Random compiler for fast Hamiltonian simulation, *Phys. Rev. Lett.* **123**, 070503 (2019).
- [38] G. Ben Arous and A. Jagannath, Spectral gap estimates in mean field spin glasses, *Commun. Math. Phys.* **361**, 1 (2018).
- [39] A. M. Childs and T. Li, Efficient simulation of sparse Markovian quantum dynamics, *Quantum Inf. Comput.* **17**, 901 (2017).
- [40] A. M. Childs and R. Kothari, Limitations on the simulation of non-sparse Hamiltonians, *Quantum Inf. Comput.* **10**, 669 (2010).
- [41] S. Adhikari and R. Baer, Stochastically bundled dissipators for the quantum master equation, *J. Chem. Theory Comput.* **21**, 4142 (2025).
- [42] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, UK, 2010).
- [43] J. Nie, W. Zi, and X. Sun, Quantum circuit for multi-qubit Toffoli gate with optimal resource, [arXiv:2402.05053](https://arxiv.org/abs/2402.05053).
- [44] L. Grover and T. Rudolph, Creating superpositions that correspond to efficiently integrable probability distributions, [arXiv:quant-ph/0208112](https://arxiv.org/abs/quant-ph/0208112).
- [45] X. Sun, G. Tian, S. Yang, P. Yuan, and S. Zhang, Asymptotically optimal circuit depth for quantum state preparation and general unitary synthesis, *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **42**, 3301 (2023).
- [46] T. G. Draper, Addition on a quantum computer, [arXiv:quant-ph/0008033](https://arxiv.org/abs/quant-ph/0008033).
- [47] M. M. Wilde, *Quantum Information Theory* (Cambridge University Press, USA, 2013), 1st ed.