

# **Early fault-tolerant quantum algorithms for Hamiltonian simulation: benchmarking and applications**

Thesis submitted in partial fulfillment  
of the requirements for the degree of

*Master of Science in Computer Science and Engineering by Research*

by

Zeeshan Ahmed  
2019111024

[zeeshan.ahmed@research.iiit.ac.in](mailto:zeeshan.ahmed@research.iiit.ac.in)



The Centre for Quantum Science and Technology (CQST)  
International Institute of Information Technology (IIIT)  
Hyderabad - 500 032, INDIA  
January 2025

Copyright © Zeeshan Ahmed, 2025  
All Rights Reserved

International Institute of Information Technology  
Hyderabad, India

## CERTIFICATE

It is certified that the work contained in this thesis, titled “Early fault-tolerant quantum algorithms for Hamiltonian simulation: benchmarking and applications” by Zeeshan Ahmed, has been carried out under my supervision and is not submitted elsewhere for a degree.

---

Date

---

Adviser: Prof. Shantanav Chakraborty

To No One

“A computer would deserve to be called intelligent if it could deceive a human into believing  
that it was human”

# Abstract

Classical computers face fundamental barriers when simulating quantum phenomena due to the exponential growth of quantum state spaces. This limitation motivates quantum simulation, particularly Hamiltonian simulation, as a central application for quantum computers. Simulating quantum dynamics is essential for studying materials and molecules, and Hamiltonian simulation is an integral subroutine of important quantum algorithms including ground state preparation, quantum phase estimation, quantum linear systems, and quantum singular value transformation. While various Hamiltonian simulation methods have been developed, those offering optimal complexity such as qubitization and LCU-based methods require oracular assumptions, complicated multi-qubit control, and extensive ancilla qubits, making them unsuitable for near and intermediate-term quantum computers. In contrast, hardware-friendly methods such as Trotterization, qDRIFT, and Single-Ancilla LCU appear more practical, but their performance for concrete physical systems and applications of interest remains unclear. This thesis provides comprehensive numerical benchmarks comparing Hamiltonian simulation algorithms for early fault-tolerant quantum computers. We analyze when and why different methods achieve optimal performance based on system structure, precision requirements, and hardware constraints. We benchmark Single-Ancilla LCU (SA-LCU), first and second-order Trotterization, and qDRIFT across two distinct classes of quantum systems. For sparse Ising chains with nearest-neighbor interactions, first-order Trotterization achieves better performance at moderate precision due to tight commutator bounds, while SA-LCU dominates at high precision owing to its polylogarithmic scaling. For dense quantum chemistry Hamiltonians, SA-LCU consistently achieves the lowest gate counts across nearly all regimes. These results show that no single algorithm dominates universally, optimal method selection depends on Hamiltonian structure and target precision. We extend this comparative framework to two applications that use Hamiltonian simulation as a subroutine. First, we consider the problem of estimating the magnetization of the ground state of the transverse-field Ising Hamiltonian. We vary the transverse-field coupling strength across a quantum phase transition and benchmark the performance of various Hamiltonian simulation procedures. Our numerical simulations show that SA-LCU outperforms other techniques, requiring shorter circuit depths while using minimal ancilla overhead. Secondly, we consider the problem of simulating open systems dynamics, an avenue where simulating Hamiltonian dynamics finds widespread applications. However, most quantum algorithms for this problem require a lot of resources, rendering them infeasible for early fault-tolerant quantum computers. We develop a collision model framework, where a single environment qubit repeatedly interacts with the underlying system. This sequence of simple interactions allows for the simulation of both Markovian and non-Markovian open systems dynamics, while naturally incorporating near-term Hamiltonian simulation techniques. Our numerical validation shows that second-order Trotter methods achieve the best asymptotic scaling for long-time evolution, while SA-LCU remains competitive for high-precision, short-time regimes. Our findings offer concrete guidance for the choice of Hamiltonian simulation method driven by the structure of the system, target precision, and available hardware resources. In the early fault-tolerant era, no single algorithm dominates universally, and understanding these trade-offs is essential.

# Acknowledgment

I would like to thank Prof. Shantanav Chakrabarty, with who I have been working together since 2022. Starting as a TA for Automata Theory, and then ending up as his dual degree student, I am grateful for his guidance in both computer science as well as in other worldly aspects. He was approving of my ventures to explore both within and outside the domain of my research, allowing me to discover and improve in several aspects of computer science. I would not be as capable as I am today without his support.

This thesis would not have been possible without Kushagra Garg. I would like to think him for being the pillar of my time here. His support in my life definitely will not go unnoticed, and I hope that his time at his PhD is as fruitful and exciting as it was here. I would like to thank my friends, Shreyas, Arjo, Rutvij, Abhishek for being the best bunch of people to do research with. I thank Pulak, Ashwin, Tanishq, Shivansh, and Anvay for all the fun trips and dinners throughout my time here at Hyderabad. I would not be so clear about my life if not for my school friends Satvik, Kaushik, and Narmin. I would also like to thank my parents for believing in me and encouraging me to be more critical about my decisions, their love and dedication to make my life better means more than anything else to me.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Fundamentals of Quantum Computing</b>	<b>6</b>
2.1	Quantum Postulates . . . . .	6
2.2	Density Matrices and Reduced States . . . . .	7
2.3	Quantum Circuits and Universal Gate Sets . . . . .	9
<b>3</b>	<b>Hamiltonian Simulation</b>	<b>12</b>
3.1	Circuit Synthesis . . . . .	13
3.2	Pauli Decomposition Model . . . . .	14
3.2.1	First-Order Trotterization . . . . .	14
3.2.2	Second-Order and Higher Formulas . . . . .	15
3.2.3	Randomized compilation: qDRIFT . . . . .	16
3.2.4	Recent Developments: THRIFT in the Interaction Picture . . . . .	17
3.3	LCU Decomposition Model . . . . .	18
3.4	Single-Ancilla LCU Framework . . . . .	19
3.4.1	Specialization to Hamiltonian simulation . . . . .	20
3.4.2	Comparison with Other Methods . . . . .	24
3.5	Benchmarking and Numerical Simulations . . . . .	24
3.5.1	Resource Trade-offs . . . . .	25
3.5.2	One-Dimensional Ising Model . . . . .	26
3.5.3	Quantum Chemistry . . . . .	27
3.5.4	Summary . . . . .	29
<b>4</b>	<b>Applications: Ground State Preparation</b>	<b>30</b>
4.1	The Single-Ancilla LCU Algorithm for Ground State Property Estimation . . . . .	31
4.2	Taylor Series Decomposition for Hamiltonian Evolution . . . . .	34
4.3	Comparative Resource Analysis . . . . .	36
4.4	Numerical Simulations . . . . .	36
4.4.1	Ising Model . . . . .	37
4.4.2	Quantum Chemistry . . . . .	38
<b>5</b>	<b>Applications: Simulating quantum collision models</b>	<b>40</b>
5.1	Introduction . . . . .	40
5.2	Simulating Markovian Collision Models . . . . .	40
5.3	Single-Ancilla LCU Implementation for Collision Models . . . . .	42
5.4	Extension to Non-Markovian Dynamics . . . . .	43
5.4.1	Memory-Retaining Collision Map . . . . .	43
5.4.2	Algorithm for Non-Markovian Dynamics . . . . .	45
5.5	Simulating Lindbladian Dynamics via Collision Models . . . . .	46
5.6	Numerical benchmarking . . . . .	47
<b>6</b>	<b>Conclusion</b>	<b>51</b>

# List of Theorems

2.2.1 Definition (Density matrix) . . . . .	8
2.3.1 Definition (Pauli gates) . . . . .	9
2.3.2 Definition (Hadamard gate) . . . . .	9
2.3.3 Definition (Controlled-NOT gate) . . . . .	10
2.3.4 Definition (Rotation gates) . . . . .	10
2.3.5 Definition (Special unitary group $SU(2)$ ) . . . . .	10
2.3.6 Theorem (Solovay–Kitaev) . . . . .	10
3.1.1 Definition (Hamiltonian) . . . . .	13
3.4.1 Theorem (Estimating expectation values of observables) . . . . .	23
3.4.2 Theorem (Single–Ancilla LCU for Hamiltonian Simulation) . . . . .	23
4.0.1 Definition (Ground State Property Estimation) . . . . .	31
4.1.1 Theorem (Ground State Property Estimation via Single–Ancilla LCU) . . . . .	31
4.2.1 Theorem (Imprecise Unitary Ground State Preparation) . . . . .	34
5.2.1 Definition (Markovian Collision Map) . . . . .	41
5.2.2 Definition (Markovian $K$ -Collision Map) . . . . .	41
5.2.3 Lemma (Bounds on the Markovian Collision Map) . . . . .	42
5.4.1 Definition (Non-Markovian Collision Map) . . . . .	44
5.4.2 Lemma (Bounds on Non-Markovian Approximate Collision Map) . . . . .	45

# List of Algorithms

1	qDRIFT randomized compiler for Hamiltonian simulation . . . . .	17
2	Standard LCU (Prepare-Select-Unprepare) . . . . .	18
3	Single-Ancilla LCU Expectation Value Estimation . . . . .	20
4	Single-Ancilla Ground State Property Estimation (One Run) . . . . .	33
5	Estimate Observable After $K$ -Collision Map Using SA-LCU . . . . .	43
6	Estimate Observable After Non-Markovian $K$ -Collision Map . . . . .	45

# List of Figures

2.1	Circuit for Ising model time evolution. Hadamard gates prepare the initial state $ ++\rangle$ . The CNOT– $R_Z(\phi)$ –CNOT sequence implements the ZZ interaction $e^{i\theta Z_1 \otimes Z_2}$ with $\phi = -2\theta = -2Jt$ . Computational basis measurements complete the simulation. . . . .	11
3.1	Standard LCU circuit implementing the prepare-select-unprepare protocol . . . . .	19
3.2	Single-ancilla LCU circuit. $V_1$ is applied controlled on $ 1\rangle$ , $V_2$ anti-controlled on $ 0\rangle$ . Measuring $X \otimes O$ yields a random outcome whose average converges to $\text{Tr}[O f(H) \rho_0 f(H)^\dagger]$ . . . . .	19
3.3	Benchmarking results for a 25-qubit one-dimensional Ising model. Left: CNOT gate count as a function of evolution time $t$ and error $\varepsilon$ . Right: Method with lowest gate count in each regime. Single-ancilla LCU dominates at high precision ( $\varepsilon < 10^{-3}$ ), while first-order Trotterization is preferable at larger error tolerances. . . . .	26
3.4	CNOT gate count versus system size $N$ for the one-dimensional Ising model at fixed $t = 1$ and $\varepsilon = 10^{-3}$ . All methods scale linearly in $N$ , with single-ancilla LCU achieving the lowest gate counts across all system sizes. . . . .	27
3.5	CNOT gate count for a 25-qubit Ising chain. Left: gate count versus evolution time $t$ at fixed $\varepsilon = 10^{-3}$ . Right: gate count versus target error $\varepsilon$ at fixed $t = 10$ . The logarithmic error scaling of single-ancilla LCU dramatically outperforms product formulas at high precision. . . . .	27
3.6	Benchmarking results for methane ( $\text{CH}_4$ ) Hamiltonian simulation. Left: CNOT gate count as a function of evolution time $t$ and error $\varepsilon$ . Right: Gate count versus time at fixed error $\varepsilon = 10^{-3}$ . Single-ancilla LCU achieves the lowest gate counts across most regimes, demonstrating clear advantages for dense, large- $L$ Hamiltonians. . . . .	28
3.7	CNOT gate count versus target error $\varepsilon$ for methane at fixed evolution time $t = 10$ . The dense structure of the quantum chemistry Hamiltonian amplifies the advantage of single-ancilla LCU: its logarithmic error scaling (red) achieves a seven-order-of-magnitude improvement over first-order Trotterization at $\varepsilon = 10^{-4}$ . qDRIFT (green) performs comparatively well due to its $\lambda$ -dependence rather than $L$ -dependence. . . . .	29
4.1	Single-ancilla LCU circuit for ground state property estimation. The ancilla (top) controls Hamiltonian evolution operators sampled from the LCU decomposition. The controlled and anti-controlled evolution operators are applied back-to-back, followed by a joint measurement of the observable $X \otimes O$ on both the ancilla and system registers. . . . .	33
4.2	CNOT gate count scaling for ground state preparation of the 25-qubit transverse-field Ising model using single-ancilla LCU with different Hamiltonian simulation backends. The single-ancilla LCU method maintains relatively low gate counts across the error range. . . . .	37
4.3	Ground state magnetization $M_z$ versus transverse field strength $h$ for 8, 10, and 12-qubit Ising chains obtained via exact diagonalization. The quantum phase transition occurs near $h \approx 1$ , with larger systems exhibiting sharper transitions toward the thermodynamic limit. . . . .	38
4.4	CNOT gate count scaling for ground state preparation of the methane molecule using single-ancilla LCU with different Hamiltonian simulation backends. The dense coupling structure of quantum chemistry Hamiltonians favors methods with reduced dependence on the number of Hamiltonian terms. . . . .	39
5.1	Schematic representation of the collision model depicting interactions between a quantum system $S$ (blue) and an ordered sequence of environmental subsystems $E_0, E_1, \dots, E_{m-1}$ (green). The system undergoes $K$ sequential collisions, where each collision involves interaction with one environmental subsystem followed by tracing out that subsystem, implementing the Markovian $K$ -collision map $\mathcal{M}_K$ . . . . .	41

5.2	Quantum circuit corresponding to each run of Algorithm 6, simulating a non-Markovian $K$ -collision map using Hamiltonian simulation by SA-LCU. The algorithm applies controlled and anti-controlled sampled unitaries ( $X_j, Y_j$ ) for the interaction between the system and each sub-environment, followed by an interaction between consecutive sub-environments using the channel ( $C_{j,j+1}$ ). This sequence is repeated for $K$ collisions. At the end of the process, the ancilla qubit and the system are measured. . . . .	46
5.3	Collision model dynamics for different environment energy levels $\omega$ (coefficient of the environment Hamiltonian $H_{E_j} = \omega 1\rangle\langle 1 $ ). Higher values of $\omega$ result in faster decay to the steady state with magnetization $M_z = 1$ , demonstrating how the environment energy parameter controls the dissipation rate. . . . .	48
5.4	CNOT gate count benchmarks for collision model Lindbladian simulation. (Left) Gate count vs. precision $\varepsilon$ for fixed evolution time $t = 1$ . (Right) Gate count vs. evolution time for fixed precision $\varepsilon = 0.01$ . . . . .	49
5.5	Effect of partial swap parameter $p$ on magnetization evolution in a 6-qubit Ising chain under non-Markovian collision model dynamics. . . . .	50

# Chapter 1

## Introduction

Classical mechanics describes the universe as a deterministic system where, in principle, complete knowledge of initial conditions allows perfect prediction of future states. In the early twentieth century, experiments on matter and radiation revealed phenomena that classical mechanics could not explain, blackbody radiation spectra, the photoelectric effect, atomic stability, and wave-particle duality being the prominent ones. These observations led to the development of quantum mechanics, a framework in which physical states exist as superpositions, measurements yield probabilistic outcomes, and the act of observation disturbs the system being measured. Several decades later, the theory of computation was formalized through the work of Turing, Church, and others. The Church-Turing thesis established that diverse computational models, including probabilistic ones, are equivalent in their computational power. The Extended Church-Turing thesis went further, asserting that any realistic model of computation can be efficiently simulated by a probabilistic Turing machine with at most polynomial overhead. This raises a natural question, can quantum mechanical processes be efficiently simulated on a classical computer?

Quantum mechanics operates by different principles than classical physics. As we moved from understanding individual quantum systems to complex many-body quantum phenomena, it became increasingly clear that classical computers face inherent limitations in simulating quantum dynamics. A system of  $n$  qubits occupies a  $2^n$ -dimensional complex Hilbert space, making direct classical simulation exponentially expensive as system size grows. The recognition that quantum systems might be better simulated by quantum computers than by classical machines motivated the field of quantum computation. Richard Feynman's 1982 observation that "nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical" articulated a fundamental insight: if the universe operates quantum mechanically, then quantum computers, machines that themselves exploit quantum phenomena, should provide natural tools for simulating quantum systems [1].

These ideas were demonstrated through landmark results showing quantum computational advantages. Grover's search algorithm [2] showed that quantum computers could search unstructured databases with quadratic speedup over classical approaches, requiring only  $O(\sqrt{N})$  queries compared to the classical  $O(N)$ . Shor's factoring algorithm [3] revealed that quantum computers could factor integers and compute discrete logarithms in polynomial time, problems believed to require exponential classical resources. These results showed that quantum computers may have different computational capabilities than their classical counterparts. However, despite decades of theoretical progress in quantum algorithms, the practical realization of these theoretical advantages remains elusive. To date, there exists no real-world problem for which a quantum computer has demonstrated a sustained, verifiable advantage over the best known classical algorithms on practical problem instances, highlighting the substantial engineering challenges remaining before quantum computing transitions from theoretical promise to practical utility.

Hamiltonian simulation directly addresses the original motivation for quantum computing. The task of simulating the time evolution of a quantum system under a given Hamiltonian  $H$ , computing  $e^{-iHt}|\psi\rangle$  for an initial state  $|\psi\rangle$ , represents precisely the kind of quantum mechanical process that Feynman envisioned quantum computers would naturally excel at. Classical computers must track exponentially many amplitudes, whereas quantum computers can embed the simulation directly into their native quantum dynamics.

This task provides the foundation for understanding many aspects of quantum physics, from molecular chemistry and materials science to high-energy physics and condensed matter phenomena. Beyond its direct applications in simulating physical systems, Hamiltonian simulation serves as a critical subroutine in many of the most important quantum algorithms. The HHL algorithm for solving linear systems of equations [4] relies on Hamiltonian simulation to implement the required unitary transformations. Ground state preparation algo-

rithms [5, 6] use Hamiltonian simulation to evolve quantum states toward the lowest energy eigenstates of target Hamiltonians, with applications ranging from quantum chemistry to condensed matter physics. Quantum phase estimation, which extracts eigenvalues of Hamiltonians, depends fundamentally on controlled time evolution operators implemented through Hamiltonian simulation. Quantum algorithms for solving differential equations [7] rely on Hamiltonian simulation to implement the required time evolution. Simulating open quantum systems dynamics, governed by Lindbladian master equations, can be reduced to Hamiltonian simulation through collision model frameworks [8], with applications to quantum thermodynamics and error mitigation. Quantum walk algorithms [9], which underlie spatial search and graph problems, are also intimately connected to Hamiltonian simulation. More broadly, quantum algorithms for linear algebra, including quantum signal processing [10, 11] and related techniques, build upon Hamiltonian simulation primitives. This widespread dependence makes the performance of Hamiltonian simulation methods critical to the practical viability of a broad class of quantum algorithms.

Despite the theoretical promise of quantum algorithms, translating these advantages into practical reality faces significant challenges. Current Noisy Intermediate-Scale Quantum (NISQ) devices, typically containing 50-1000 physical qubits with gate error rates around 0.1-0.3%, suffer from noise that fundamentally limits algorithmic depth and complexity [12, 13]. While recent demonstrations like Google’s 2024 Willow processor have achieved below-threshold error correction on 101-qubit surface codes, translating this into practical computation requires substantial overhead. Surface codes require 50-100 physical qubits per logical qubit for modest error suppression (code distances 5-7), with higher-distance codes demanding hundreds to over a thousand physical qubits for each logical qubit [14]. Alternative schemes like LDPC codes [15, 16] or tesseract codes [17] can achieve better ratios but remain in early development stages.

These resource constraints have motivated research into “early fault-tolerant quantum computing” [18, 9], an anticipated intermediate regime where quantum error correction successfully suppresses logical errors below physical thresholds, enabling deeper circuits than NISQ devices. However, error correction overhead limits the number of logical qubits to tens or low hundreds even on devices with thousands to tens of thousands of physical qubits. This intermediate stage differs from full fault-tolerant quantum computing, envisioned for the 2030s and beyond, which would provide thousands of logical qubits from millions of physical qubits with sufficient resources for large-scale applications. Early fault-tolerant devices impose specific algorithmic constraints of minimizing ancilla qubits that are often used to construct sophisticated multi-qubit controlled operations. These operations must be avoided even at the cost of additional classical repetitions. Recent work has established that the logarithmic ancilla overhead inherent to standard block-encoding approaches is provably unavoidable [19], further motivating the development of alternative techniques. These requirements motivate a detailed examination of quantum algorithms designed for early fault-tolerant devices and their comparative performance under realistic constraints, with Hamiltonian simulation at the center of this investigation.

The problem of Hamiltonian simulation has spawned diverse algorithmic approaches, each with distinct resource trade-offs. Consider an  $n$ -qubit Hamiltonian decomposed as  $H = \sum_{j=1}^L h_j$  into  $L$  terms, where each  $h_j$  can be efficiently exponentiated. The simulation task is to implement a unitary  $\tilde{U}$  approximating the time evolution  $e^{-iHt}$  for evolution time  $t$  with error  $\|\tilde{U} - e^{-iHt}\| \leq \varepsilon$ , where  $\|A\| = \max_{|\psi\rangle} \|A|\psi\rangle\| / \||\psi\rangle\|$  denotes the operator norm (spectral norm), the largest singular value of  $A$ . Algorithm complexity depends on these parameters  $n, L, t, \varepsilon$  and on norm quantities such as  $\lambda = \sum_j \|h_j\|$  (the one-norm of coefficients) characterizing the Hamiltonian structure.

Product formula methods, pioneered by Lloyd [20] and refined through Suzuki’s higher-order formulas [21], decompose the evolution into sequences of simpler unitaries  $e^{-ih_j\tau}$  corresponding to individual terms. For a  $p$ -th order product formula, the Trotter error satisfies  $\|S_p(t) - e^{-iHt}\| = O(\alpha_{\text{comm}} t^{p+1})$ , where the commutator norm  $\alpha_{\text{comm}}$  quantifies how much the individual Hamiltonian terms fail to commute with each other through nested commutator expressions [22]. Achieving error  $\varepsilon$  requires  $r = O(\alpha_{\text{comm}}^{1/p} t^{1+1/p}/\varepsilon^{1/p})$  Trotter steps. This commutator-scaling theory reduces effective simulation cost by orders of magnitude for structured Hamiltonians when terms commute or nearly commute, as in many physical lattice models. Multi-product formulas, which linearly combine multiple product formula circuits, achieve quadratic error reduction while maintaining commutator scaling [23].

Randomized approaches offer alternative resource trade-offs. The qDRIFT algorithm [24] randomly samples terms with probabilities proportional to their coefficient magnitudes, achieving gate complexity  $O((\lambda t)^2/\varepsilon)$  that is independent of the number of terms  $L$ , making it efficient for dense Hamiltonians with many small-coefficient terms. Recent work has established tighter bounds scaling linearly rather than quadratically in  $\lambda$  for certain regimes [25]. The higher-order extension qSWIFT [26] improves precision scaling from  $O(1/\varepsilon)$  to  $O(1/\varepsilon^{1/k})$  for  $k$ -th order, achieving gate complexity  $O((\lambda t)^{1+1/k}/\varepsilon^{1/k})$  and yielding up to 1000-fold gate count reductions for high-precision applications. THRIFT [27] exploits perturbative structure when Hamiltonians decompose into a dominant base term plus a small perturbation.

Linear Combination of Unitaries (LCU) methods take a fundamentally different approach, representing operators as weighted sums of unitaries and implementing these decompositions through prepare-select-unprepare circuits [28]. Standard LCU constructions require  $O(\log L)$  ancilla qubits and multi-qubit controlled operations. Qubitization and quantum signal processing [11, 10] achieve optimal query complexity  $O(\lambda t + \log(1/\varepsilon))$  to the block encoding oracle. However, each query to the block encoding incurs gate depth  $O(L)$  from the SELECT oracle, yielding end-to-end gate complexity  $O(L(\lambda t + \log(1/\varepsilon)))$  for  $L$ -term Hamiltonians.

The single-ancilla LCU (SA-LCU) framework [9], developed specifically for early fault-tolerant quantum computers, offers a simplified alternative that requires exactly one ancilla qubit and no multi-qubit controlled operations, employing randomized sampling rather than coherent state preparation. For Hamiltonian simulation, SA-LCU achieves overall gate complexity  $O(\lambda^2 t^2 \|O\|^2 / \varepsilon^2)$  through  $O(\|O\|^2 / \varepsilon^2)$  repetitions of short-depth circuits with per-run gate depth scaling as  $O(\lambda^2 t^2 \log(\lambda t \|O\| / \varepsilon) / \log \log(\lambda t \|O\| / \varepsilon))$ , where  $O$  is the observable being estimated. While the theoretical complexities of these methods have been extensively analyzed, systematic empirical comparison across diverse physical systems has been limited, leaving questions about which algorithms perform best under realistic problem conditions.

Understanding why practical performance diverges from theoretical predictions requires examining the limitations of asymptotic analysis. Asymptotic complexity analysis provides theoretical guidance but may not predict practical performance for several reasons:

- **Prefactors and Hidden Constants.** Big-O notation suppresses constant factors that can differ by orders of magnitude between algorithms. An algorithm with  $O(L \cdot f(\varepsilon))$  scaling may outperform one with  $O(\lambda \cdot g(\varepsilon))$  scaling when the prefactors and functional forms favor the former, even if the latter has “better” asymptotic dependence.
- **Problem-Specific Structure.** Real Hamiltonians possess structure, sparsity, locality, symmetries, commutation relations, that algorithms exploit differently. Product formulas benefit substantially from tight commutator bounds in local lattice models, while norm-based methods (qDRIFT, SA-LCU) excel when coefficient norms are small relative to the number of terms. Asymptotic analysis alone cannot reveal which structural features dominate for specific problem classes.
- **Crossover Regimes.** Different algorithms dominate in different parameter regimes (system size, evolution time, precision requirements). A method optimal for high-precision long-time evolution may be suboptimal for moderate-precision short-time simulation. Identifying these crossover boundaries requires explicit numerical evaluation.

These gaps between theoretical analysis and practical performance motivate comprehensive numerical benchmarks and comparative studies that establish empirically when and why different Hamiltonian simulation methods achieve optimal performance for early fault-tolerant quantum computers, forming the topic of this thesis. Through numerical benchmarking and comparative analysis, we provide a systematic understanding of algorithm performance for Hamiltonian simulation on early fault-tolerant quantum computers. Our central contributions are empirical rather than purely theoretical.

We develop a systematic benchmarking methodology that evaluates algorithms across multiple dimensions. First, we analyze circuit depth per coherent run, focusing on gate count and particularly CNOT gates as proxies for error-prone two-qubit operations. Second, we quantify classical repetition requirements, measuring the sample complexity needed for achieving target precision. Third, we account for ancilla qubit overhead to determine total qubit requirements. Finally, we characterize scaling behavior across evolution time  $t$ , error tolerance  $\varepsilon$ , system size  $N$ , and Hamiltonian structure parameters including the number of terms  $L$  and coefficient norm  $\lambda$ .

This framework enables direct, quantitative comparison of SA-LCU, first and second-order Trotterization, and qDRIFT under consistent conditions, revealing performance characteristics that asymptotic analysis alone cannot capture. We apply this methodology to sparse Hamiltonians through one-dimensional transverse-field Ising model benchmarks. The transverse-field Ising Hamiltonian is defined as  $H = -\sum_{\langle i,j \rangle} Z_i Z_j - h \sum_i X_i$ , where  $Z_i$  and  $X_i$  are Pauli-Z and Pauli-X operators acting on the  $i$ -th qubit,  $\langle i,j \rangle$  denotes summation over nearest-neighbor pairs, and  $h$  is the transverse magnetic field strength. The first term represents classical ferromagnetic coupling between adjacent spins, favoring aligned states in the  $z$ -basis. The second term introduces quantum fluctuations through a transverse field in the  $x$ -direction, creating non-commuting terms that give the Hamiltonian its quantum character. Throughout our benchmarks, we measure the average magnetization  $M_z = \frac{1}{N} \langle \sum_{i=1}^N Z_i \rangle$ , which represents the mean magnetic moment across all spins in the system. This observable is particularly well-suited for the transverse-field Ising model serving as the parameter characterizing the quantum phase transition, transitioning from the ferromagnetic phase to the paramagnetic phase as the transverse field increases. It also provides a way to validate against exact diagonalization results which can be calculated

numerically for reasonable sized system. For open quantum system simulations, it captures how decoherence affects the system over time through its characteristic decay.

This system and the observable together form a model that serve as an ideal benchmark system due to its sparse, local interaction structure with only nearest-neighbor couplings, well-characterized quantum phase transition at  $h \approx 1$  for the one-dimensional chain, and analytical tractability for validation while maintaining physical relevance to quantum many-body systems. Our benchmark results reveal that second-order Trotterization achieves the lowest gate counts in the lower-precision regime due to tight commutator bounds, while SA-LCU exhibits crossover behavior and becomes the preferred method at higher precision where its  $O(\log(1/\varepsilon)/\log\log(1/\varepsilon))$  per-run depth scaling yields substantial improvements over Trotter's  $O(1/\varepsilon)$  dependence. qDRIFT demonstrates no advantage for sparse Hamiltonians of this type, as both the coefficient norm  $\lambda$  and the number of terms  $L$  scale identically as  $O(N)$  in nearest-neighbor systems. Beyond Hamiltonian simulation, this thesis extends the single-ancilla methodology to ground state property estimation for quantum many-body and molecular systems, as well as open quantum system simulation through collision models encompassing both Markovian and non-Markovian dynamics, demonstrating that resource advantages observed in fundamental algorithm benchmarks translate to practically relevant quantum computing tasks.

Electronic structure problems in quantum chemistry provide a contrasting benchmark regime. The Jordan-Wigner transformation [29] maps fermionic creation and annihilation operators to qubit operators, producing Hamiltonians whose term count scales rapidly with the number of spin-orbitals [30]. This dense structure differs fundamentally from the Ising model, while a 25-qubit Ising chain contains approximately 50 nearest-neighbor terms, an 18-qubit methane molecule generates nearly 7,000 Pauli terms. The parity strings required by the Jordan-Wigner encoding create predominantly non-local, Z-heavy interactions. Crucially, the coefficient norm  $\lambda$  grows more slowly than  $L$ , yielding a favorable  $\lambda/L$  ratio compared to order unity for sparse lattice models.

This Hamiltonian structure fundamentally alters the algorithmic performance landscape. Methods with complexity depending on  $L$  incur substantial overhead, while methods scaling with  $\lambda$  become highly competitive. Our methane benchmarks demonstrate that SA-LCU consistently achieves lower gate counts than first-order Trotterization across most  $(t, \varepsilon)$  regimes, with advantages exceeding  $10\times$  at high precision. qDRIFT also significantly outperforms product formulas due to its L-independent gate complexity. Second-order Trotterization struggles despite its improved asymptotic scaling, as the dense, non-local term structure yields large nested commutator norms that weaken higher-order error cancellations [22]. At representative parameters  $\varepsilon = 10^{-3}$  and  $t = 100$ , SA-LCU requires approximately  $10^{12}$  CNOT gates compared to  $10^{13}$  for first-order Trotter and  $10^{14}$  for second-order Trotter.

Beyond simulating time evolution, Hamiltonian simulation serves as a primitive for ground state property estimation. The ground state is the lowest energy eigenstate of a quantum system, and at thermal equilibrium, systems predominantly occupy this state when electronic energy gaps far exceed thermal fluctuations. This makes ground state properties the dominant contributors to physical observables at room temperature in molecular systems [30]. Ground state energies and wavefunctions determine molecular binding energies, reaction barriers, and material properties. Classical exact diagonalization requires exponential resources in system size, motivating quantum approaches. Filtering-based methods offer a path forward, with convergence rates depending on the spectral gap  $\Delta$  between ground and first excited states [31]. We present numerical demonstrations of SA-LCU with Taylor series decomposition for ground state filtering on the transverse-field Ising model, comparing against Trotterization-based implementations. Our results examine magnetization across the quantum phase transition, showing agreement with exact diagonalization and revealing crossover behavior in gate counts depending on spectral gap and precision requirements. These results demonstrate SA-LCU's viability as a subroutine within larger algorithmic protocols.

Real quantum systems inevitably interact with their environments through photon emission, magnetic field fluctuations, and other uncontrolled degrees of freedom. These interactions cause loss of coherence in the system as well as energy exchange with the environment. Even with quantum error correction, early fault-tolerant devices will operate with logical non-negligible error rates and finite coherence times, making environmental effects significant for algorithmic timescales [18], these scenarios are modeled by open system dynamics. Understanding and simulating them would be crucial for modeling realistic devices, designing error mitigation strategies, and applications in quantum thermodynamics where dissipation plays a fundamental physical role. Open system dynamics are described by Lindbladian evolution, which operates on density matrices rather than state vectors and involves non-unitary transformations. Our focus is on collision models that provide a framework for simulating these dynamics using only unitary operations on an extended Hilbert space [32, 33]. The environment is modeled as a stream of ancilla qubits, each briefly interacting with the system before being discarded. A single collision consists of applying a system-bath interaction unitary for time  $\tau$ . After  $K$  collisions with  $K\tau = t$ , the discrete dynamics approximate continuous Lindbladian evolution with error  $O(\tau^2)$ . Common decoherence channels like amplitude damping and phase damping emerge naturally from specific interactions

and bath preparations.

Each collision requires Hamiltonian simulation of the coupling, directly connecting open system simulation to the methods developed in this thesis. For early fault-tolerant devices with limited ancilla budgets, collision models offer advantage as they require only one bath ancilla at a time, use standard circuit primitives, and allow trading classical repetitions for circuit depth through method selection. We validate convergence for single-qubit amplitude damping against exact solutions, demonstrating  $O(\tau^2)$  scaling. Multi-qubit demonstrations on 5- and 10-qubit Ising chains under site-wise damping track magnetization decay from initial ferromagnetic order toward thermal equilibrium. Systematic comparisons across first-order Trotter, qDRIFT, SA-LCU, and second-order Trotter reveal that second-order Trotter achieves the best long-time scaling, while SA-LCU remains competitive for high-precision short-time evolution.

The remainder of this thesis develops these ideas systematically. Chapter 2 introduces the fundamental postulates of quantum mechanics and the basics of the circuit model of quantum computation, including the qubit formalism and quantum gates.

Chapter 3 presents the core Hamiltonian simulation techniques and our comprehensive numerical benchmarks. We systematically introduce product formula methods, qDRIFT, and SA-LCU before presenting detailed simulation results on one-dimensional Ising chains and quantum chemistry systems. The extensive benchmark plots and comparative analysis in this chapter constitute the thesis’s central contribution, revealing regime-dependent performance characteristics and establishing empirical guidelines for algorithm selection on early fault-tolerant devices.

Chapter 4 applies SA-LCU to ground state property estimation problem. We present simulation results on the transverse-field Ising model, validating the single-ancilla approach with Taylor series and comparing against Trotterization-based implementations. The numerical demonstrations illustrate SA-LCU’s viability as a subroutine in complex algorithmic protocols.

Chapter 5 presents frameworks for simulating open quantum dynamics through collision models. We provide comprehensive numerical validation on Ising spin chains under amplitude damping, including convergence studies, and systematic gate count comparisons across different Hamiltonian simulation methods. These simulation results establish collision models as practical frameworks for near-term open system simulation and reveal method-dependent performance trade-offs based on time scales and precision requirements.

Chapter 6 concludes the thesis by summarizing our contributions, discussing the broader implications and limitations of our work, and identifying open problems for future research.

This thesis demonstrates through extensive numerical evidence that algorithm performance depends critically on problem structure, precision requirements, and hardware constraints. These are the factors that asymptotic analysis alone cannot capture. Our comprehensive benchmarks and comparative studies provide the empirical foundation for informed algorithm selection in the early fault-tolerant quantum computing era.

## Chapter 2

# Fundamentals of Quantum Computing

By the mid-1920s, quantum mechanics existed in two seemingly incompatible formulations: Heisenberg's matrix mechanics and Schrödinger's wave mechanics. Von Neumann's work from 1927–1932 unified these approaches by showing both are representations of an abstract Hilbert space framework, providing the rigorous mathematical foundation that remains standard today [34, 35].

Roughly fifty years elapsed before this formalism was connected to computation. In 1980, Benioff demonstrated that classical Turing machines could be embedded in quantum mechanical Hamiltonians, establishing that quantum systems could in principle perform computation [36]. Feynman's 1982 observation that classical computers appear fundamentally incapable of efficiently simulating quantum phenomena, while a quantum computer could do so naturally, is conventionally taken as the birth of quantum computing [1]. Deutsch formalized this vision in 1985 by defining the universal quantum Turing machine, establishing that quantum mechanics permits a computational model at least as powerful as its classical counterpart [37]. Although the postulates describe continuous-time unitary evolution, Deutsch's 1989 work introduced quantum computational networks, providing a discrete circuit framework more amenable to algorithm design [38]. The Solovay–Kitaev theorem then established that any unitary operation can be approximated to arbitrary precision using only a small universal gate set, with the number of gates scaling polylogarithmically in the inverse error [39, 40]. These developments justify our use of the circuit model throughout this thesis, built upon the following quantum postulates.

## 2.1 Quantum Postulates

**Postulate 1: State Space.** The state of an isolated quantum system is described by a unit vector  $|\psi\rangle$  in a complex Hilbert space  $\mathcal{H}$ , satisfying the normalization condition  $\|\psi\|^2 = \langle\psi|\psi\rangle = 1$ . Two vectors differing only by a global phase represent the same physical state.

A *state* is the complete physical description of a quantum system at a given instant. For a single qubit, the simplest quantum system, the Hilbert space is  $\mathcal{H} = \mathbb{C}^2$  and this state takes the form

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad (2.1.1)$$

where  $\alpha, \beta \in \mathbb{C}$  satisfy the normalization condition  $\langle\psi|\psi\rangle = |\alpha|^2 + |\beta|^2 = 1$ . The computational basis states  $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$  and  $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$  span the space, and the complex amplitudes  $\alpha, \beta$  encode what's known as the relative phases that determine interference effects. The equivalence of states differing by a *global phase*  $e^{i\theta}$  follows directly from how physical predictions arise from the Hilbert space structure that will be better understood in context of postulate 3.

**Postulate 2: Evolution.** The evolution of a closed quantum system is described by a unitary operator. If the system is in state  $|\psi(t_1)\rangle$  at time  $t_1$ , it evolves to  $|\psi(t_2)\rangle = U(t_1, t_2)|\psi(t_1)\rangle$  at time  $t_2$ , where  $U^\dagger U = I$ .

This postulate connects to the Schrödinger equation, which governs continuous-time evolution:

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = H |\psi(t)\rangle, \quad (2.1.2)$$

where  $H$  is the Hamiltonian (a Hermitian operator representing the system's total energy). Integrating this differential equation yields

$$|\psi(t)\rangle = e^{-iHt/\hbar} |\psi(0)\rangle. \quad (2.1.3)$$

The operator  $U(t) = e^{-iHt/\hbar}$  is unitary precisely because  $H$  is Hermitian:  $U^\dagger U = e^{iHt/\hbar} e^{-iHt/\hbar} = I$ . Thus the Schrödinger equation guarantees that time evolution preserves the norm of quantum states:  $\langle \psi | U^\dagger U | \psi \rangle = \langle \psi | \psi \rangle = 1$ .

**Postulate 3: Measurement.** Quantum measurements are described by a collection of measurement operators  $\{M_m\}$  satisfying the completeness relation  $\sum_m M_m^\dagger M_m = I$ . If the system is in state  $|\psi\rangle$ , outcome  $m$  occurs with probability  $p(m) = \langle \psi | M_m^\dagger M_m | \psi \rangle$ , and the post-measurement state is  $M_m |\psi\rangle / \sqrt{p(m)}$ .

Each measurement operator  $M_m$  represents one possible measurement outcome. The completeness relation  $\sum_m M_m^\dagger M_m = I$  ensures that probabilities sum to unity:

$$\sum_m p(m) = \sum_m \langle \psi | M_m^\dagger M_m | \psi \rangle = \langle \psi | \left( \sum_m M_m^\dagger M_m \right) | \psi \rangle = \langle \psi | \psi \rangle = 1. \quad (2.1.4)$$

Measurement disturbs the system, leaving it in the new normalized state  $M_m |\psi\rangle / \sqrt{p(m)}$ . This formalism clarifies why global phase is physically irrelevant: if  $|\psi\rangle$  and  $e^{i\theta} |\psi\rangle$  differ only by a global phase, all measurement probabilities are identical since  $\langle \psi | e^{-i\theta} M_m^\dagger M_m e^{i\theta} | \psi \rangle = \langle \psi | M_m^\dagger M_m | \psi \rangle$ . Global phase factors cancel in all physical predictions, so states differing by global phase are indistinguishable.

**Postulate 4: Composite Systems.** The state space of a composite physical system is the tensor product of the component state spaces:  $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$ . If system  $A$  is in state  $|\psi\rangle_A$  and system  $B$  is in state  $|\phi\rangle_B$ , the joint state is  $|\Psi\rangle_{AB} = |\psi\rangle_A \otimes |\phi\rangle_B$ .

The tensor product is a bilinear operation: for  $|\psi\rangle = \sum_i \alpha_i |i\rangle_A$  and  $|\phi\rangle = \sum_j \beta_j |j\rangle_B$ ,

$$|\psi\rangle \otimes |\phi\rangle = \sum_{i,j} \alpha_i \beta_j |i\rangle_A \otimes |j\rangle_B, \quad (2.1.5)$$

where we often abbreviate  $|i\rangle_A \otimes |j\rangle_B$  as  $|i,j\rangle$  or  $|ij\rangle$ . Operators act locally via  $(A \otimes B)(|\psi\rangle \otimes |\phi\rangle) = (A|\psi\rangle) \otimes (B|\phi\rangle)$ .

The tensor product structure has dimension  $\dim(\mathcal{H}_A \otimes \mathcal{H}_B) = \dim(\mathcal{H}_A) \times \dim(\mathcal{H}_B)$ . For two qubits,  $\mathbb{C}^2 \otimes \mathbb{C}^2 = \mathbb{C}^4$ , with basis  $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ . This multiplicative scaling is physically motivated since local operations performed independently on subsystems combine as  $U_A \otimes U_B$ , and for uncorrelated product states, measurement probabilities factorize as  $p_{AB}(i,j) = p_A(i) \cdot p_B(j)$ .

The tensor product structure allows not all states in  $\mathcal{H}_A \otimes \mathcal{H}_B$  to be written as product states  $|\psi\rangle_A \otimes |\phi\rangle_B$ . States that cannot be factored in this way are called *entangled*. For example, the Bell state  $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$  cannot be decomposed into a product of single-qubit states. To see this, suppose  $|\Phi^+\rangle = (\alpha|0\rangle + \beta|1\rangle) \otimes (\gamma|0\rangle + \delta|1\rangle)$ . Expanding gives coefficients  $\alpha\gamma, \alpha\delta, \beta\gamma, \beta\delta$  for  $|00\rangle, |01\rangle, |10\rangle, |11\rangle$  respectively. Matching to  $|\Phi^+\rangle$  requires  $\alpha\gamma = \beta\delta = 1/\sqrt{2}$  and  $\alpha\delta = \beta\gamma = 0$ . The latter conditions force  $\alpha = 0$  or  $\delta = 0$ , but either choice contradicts the former conditions. This algebraic obstruction is the origin of entanglement, and the tensor product is the minimal structure that permits such correlations while respecting locality. Entangled states exhibit correlations that cannot be replicated by a classical system, enabling phenomena such as quantum teleportation and violations of Bell inequalities [41, 35].

## 2.2 Density Matrices and Reduced States

The pure state formalism of Postulate 1 suffices for isolated systems, but entanglement introduces a fundamental limitation: when a system is entangled with another, no pure state can describe the subsystem alone. This section develops the density matrix formalism, which extends quantum mechanics to handle subsystems of entangled states, classical uncertainty, and open quantum systems.

Consider the Bell state  $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$  shared between Alice (qubit  $A$ ) and Bob (qubit  $B$ ). The joint system is in a pure state with complete quantum mechanical description. But what is the state of Alice's qubit alone?

Suppose Alice's qubit could be described by the pure state  $|\psi\rangle_A = \alpha|0\rangle + \beta|1\rangle$ . Measuring in the computational basis, Alice observes outcome 0 or 1 each with probability 1/2 (since tracing over Bob's qubit in  $|\Phi^+\rangle$  gives equal weight to both outcomes). This rules out  $|\psi\rangle_A = |0\rangle$  or  $|\psi\rangle_A = |1\rangle$ , which would predict deterministic outcomes. One might guess  $|\psi\rangle_A = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle$ , but this predicts deterministic outcome + when measuring in the  $\{|+\rangle, |-\rangle\}$  basis, whereas Alice actually observes + or - each with probability 1/2. In fact, Alice obtains uniformly random outcomes in every measurement basis. No pure state  $|\psi\rangle_A$  reproduces

this behavior: pure states always have at least one basis in which the outcome is deterministic. The density matrix formalism resolves this limitation.

**Definition 2.2.1** (Density matrix). *A density operator (or density matrix)  $\rho$  on a Hilbert space  $\mathcal{H}$  is a positive semidefinite operator with unit trace. Density matrices satisfy three properties:*

- Hermitian:  $\rho = \rho^\dagger$
- Positive semidefinite:  $\langle \phi | \rho | \phi \rangle \geq 0$  for all  $|\phi\rangle$
- Unit trace:  $\text{Tr}(\rho) = 1$

The trace  $\text{Tr}(\rho) = \sum_i \langle i | \rho | i \rangle$  is the sum of diagonal elements in any orthonormal basis  $\{|i\rangle\}$ ; it is basis-independent and satisfies the cyclic property  $\text{Tr}(AB) = \text{Tr}(BA)$ .

For a pure state  $|\psi\rangle$ , the corresponding density matrix is  $\rho = |\psi\rangle\langle\psi|$ . More generally, a *mixed state* represents statistical uncertainty about which pure state the system occupies: if the system is in state  $|\psi_i\rangle$  with probability  $p_i$ , the density matrix is

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|, \quad (2.2.1)$$

where  $p_i \geq 0$  and  $\sum_i p_i = 1$ . Measurement probabilities generalize naturally: outcome  $m$  occurs with probability  $p(m) = \text{Tr}(M_m^\dagger M_m \rho)$ , which reduces to the pure state formula when  $\rho = |\psi\rangle\langle\psi|$ .

The *purity*  $\text{Tr}(\rho^2)$  distinguishes pure from mixed states:  $\text{Tr}(\rho^2) = 1$  if and only if  $\rho$  is pure. For mixed states,  $\text{Tr}(\rho^2) < 1$ . The maximally mixed state of a single qubit is  $\rho = I/2$ , with purity  $\text{Tr}(\rho^2) = 1/2$ , representing complete ignorance about the qubit's state [35].

The *partial trace* extracts the state of a subsystem from a composite system. For a bipartite density matrix  $\rho_{AB}$  on  $\mathcal{H}_A \otimes \mathcal{H}_B$ , the reduced density matrix for subsystem  $A$  is

$$\rho_A = \text{Tr}_B(\rho_{AB}) = \sum_j (I_A \otimes \langle j |_B) \rho_{AB} (I_A \otimes |j\rangle_B), \quad (2.2.2)$$

where  $\{|j\rangle_B\}$  is any orthonormal basis for  $\mathcal{H}_B$ . The result is basis-independent. The reduced state  $\rho_A$  completely determines all measurement statistics for observables acting only on subsystem  $A$ .

Returning to the Bell state  $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ , we compute Alice's reduced state explicitly. The joint density matrix is

$$\rho_{AB} = |\Phi^+\rangle\langle\Phi^+| = \frac{1}{2}(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|). \quad (2.2.3)$$

Tracing over  $B$  using the basis  $\{|0\rangle_B, |1\rangle_B\}$ :

$$\begin{aligned} \rho_A &= \langle 0 |_B \rho_{AB} | 0 \rangle_B + \langle 1 |_B \rho_{AB} | 1 \rangle_B \\ &= \frac{1}{2} |0\rangle\langle 0|_A + \frac{1}{2} |1\rangle\langle 1|_A = \frac{I}{2}. \end{aligned} \quad (2.2.4)$$

The off-diagonal terms  $|00\rangle\langle 11|$  and  $|11\rangle\langle 00|$ , which encode the entanglement, vanish under the partial trace. Alice's reduced state is the maximally mixed state, confirming that no pure state can describe her qubit: the density matrix  $I/2$  predicts uniform randomness in every measurement basis, exactly as required.

The density matrix formalism reveals a crucial distinction between quantum superposition and classical ignorance. Consider two single-qubit states with identical computational basis statistics:

**State 1: Pure superposition.** The state  $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$  has density matrix

$$\rho_+ = |+\rangle\langle+| = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}. \quad (2.2.5)$$

Measuring in the computational basis yields  $P(0) = P(1) = 1/2$ . But measuring in the  $\{|+\rangle, |-\rangle\}$  basis yields  $P(+)=1$  deterministically.

**State 2: Classical mixture.** A 50/50 classical mixture of  $|0\rangle$  and  $|1\rangle$  has density matrix

$$\rho_{\text{mix}} = \frac{1}{2} |0\rangle\langle 0| + \frac{1}{2} |1\rangle\langle 1| = \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \frac{I}{2}. \quad (2.2.6)$$

Measuring in the computational basis yields  $P(0) = P(1) = 1/2$ . But measuring in the  $\{|+\rangle, |-\rangle\}$  basis also yields  $P(+)=P(-)=1/2$ .

The difference lies in the off-diagonal elements, called *coherences*. The pure state  $\rho_+$  has nonzero coherences that encode quantum interference; the mixed state  $\rho_{\text{mix}}$  has none. This distinction is experimentally observable: the two states produce different statistics in non-computational bases. The reduced state of an entangled qubit is  $I/2$ , not  $|+\rangle$ , which is why entanglement cannot be mimicked by local superpositions [35].

## 2.3 Quantum Circuits and Universal Gate Sets

The discrete circuit model provides a practical framework for describing quantum algorithms. Rather than specifying continuous-time Hamiltonian evolution, we decompose computations into sequences of elementary gates from a finite set. This section defines the fundamental gates, shows how they act on both pure and mixed states, introduces the essential two-qubit CNOT gate, and establishes universality through the Solovay–Kitaev theorem.

**Definition 2.3.1** (Pauli gates). *The Pauli gates are three fundamental single-qubit operations:*

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \quad (2.3.1)$$

These gates have the following actions on computational basis states:

- $X$  (bit-flip):  $X|0\rangle = |1\rangle$ ,  $X|1\rangle = |0\rangle$
- $Y$  (bit and phase flip):  $Y|0\rangle = i|1\rangle$ ,  $Y|1\rangle = -i|0\rangle$
- $Z$  (phase-flip):  $Z|0\rangle = |0\rangle$ ,  $Z|1\rangle = -|1\rangle$

The Pauli gates satisfy  $X^2 = Y^2 = Z^2 = I$  and anticommute pairwise:  $XY = -YX = iZ$ .

**Definition 2.3.2** (Hadamard gate). *The Hadamard gate is the single-qubit unitary operation*

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Its action on computational basis states creates equal superpositions:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \quad (2.3.2)$$

$$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle). \quad (2.3.3)$$

The Hadamard gate is self-inverse ( $H^2 = I$ ) and transforms between the computational basis and the diagonal basis  $\{|+\rangle, |-\rangle\}$ .

For pure states, a gate  $U$  acts by  $|\psi\rangle \mapsto U|\psi\rangle$ . For density matrices representing mixed states, the evolution is

$$\rho \mapsto U\rho U^\dagger. \quad (2.3.4)$$

This transformation preserves the trace ( $\text{Tr}(U\rho U^\dagger) = \text{Tr}(\rho) = 1$ ) and positive semidefiniteness, ensuring that valid density matrices remain valid after gate application. As an example, consider the  $X$  gate acting on the pure state  $\rho_+ = |+\rangle\langle +|$ :

$$X\rho_+X^\dagger = X|+\rangle\langle +|X = |-\rangle\langle -, \quad (2.3.5)$$

where we used  $X|+\rangle = |-\rangle$ . The bit-flip transforms  $|+\rangle$  to  $|-\rangle$ , as expected from  $X = HZH$ .

**Definition 2.3.3** (Controlled-NOT gate). *The controlled-NOT (**CNOT**) gate is the two-qubit entangling operation:*

$$\text{CNOT} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \quad (2.3.6)$$

Its action on computational basis states is  $\text{CNOT}|a,b\rangle = |a, a \oplus b\rangle$ , where  $\oplus$  denotes addition modulo 2. The first qubit (control) determines whether the second qubit (target) is flipped. CNOT generates entanglement from product states: for example,

$$\text{CNOT}(|+\rangle \otimes |0\rangle) = \text{CNOT}\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle\right) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = |\Phi^+\rangle, \quad (2.3.7)$$

producing the Bell state  $|\Phi^+\rangle$ . Combined with single-qubit gates, CNOT enables universal quantum computation [35].

**Definition 2.3.4** (Rotation gates). *The rotation gates about the Pauli axes are parameterized single-qubit unitaries:*

$$R_Z(\theta) = e^{-i\theta Z/2} = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}. \quad (2.3.8)$$

Similarly,  $R_X(\theta) = e^{-i\theta X/2}$  and  $R_Y(\theta) = e^{-i\theta Y/2}$ . Note that  $S = R_Z(\pi/2)$  and  $T = R_Z(\pi/4)$  up to global phase.

It had established that simple gate sets like  $\{\text{CNOT}\} \cup \text{U}(2)$  are *universal*: any unitary operation can be decomposed into these elementary gates [42]. However, an obstacle remained. Classical universal gate sets (such as AND and NOT) can implement any Boolean function *exactly*. Quantum gates are different: single-qubit unitaries form a continuum, parameterized by continuous rotation angles, and no finite gate set can reach all of them exactly. Real devices and fault-tolerant architectures supply only a finite gate set  $\mathcal{G}$ , closed under inverses. An arbitrary target unitary  $U$  must therefore be *approximated* by a finite product of gates from  $\mathcal{G}$ . How many gates are needed to achieve accuracy  $\varepsilon$ ? Naive estimates suggested the overhead might scale as  $1/\varepsilon$  or worse, potentially requiring millions of gates for the precision needed in practical algorithms. Such scaling would render quantum computation infeasible even in principle.

In 1995, Robert Solovay announced a remarkable result, any single-qubit gate can be approximated to accuracy  $\varepsilon$  using only  $\mathcal{O}(\log^c(1/\varepsilon))$  gates from a universal finite set. The scaling is *polylogarithmic*, achieving accuracy  $\varepsilon = 10^{-6}$  requires roughly 60 gates rather than millions. Kitaev independently established the result in 1997, generalizing it to  $\text{SU}(d)$  for arbitrary dimensions [39]. The theorem's constructive proof exploits the fact that commutators of group elements close to the identity can be approximated more efficiently than expected, enabling recursive refinement of rough initial approximations [40].

**Definition 2.3.5** (Special unitary group  $\text{SU}(2)$ ). *The special unitary group  $\text{SU}(2)$  is the group of  $2 \times 2$  complex unitary matrices with determinant 1:*

$$\text{SU}(2) = \{U \in \mathbb{C}^{2 \times 2} : U^\dagger U = I \text{ and } \det(U) = 1\}.$$

Every element  $U \in \text{SU}(2)$  can be parameterized as

$$U = \begin{pmatrix} \alpha & -\beta^* \\ \beta & \alpha^* \end{pmatrix}$$

where  $\alpha, \beta \in \mathbb{C}$  satisfy  $|\alpha|^2 + |\beta|^2 = 1$ . The group  $\text{SU}(2)$  represents single-qubit operations modulo global phase.

Physical single-qubit operations lie in  $\text{U}(2)$ , but global phases are physically irrelevant, so one typically works modulo  $\text{U}(1)$  and identifies single-qubit dynamics with  $\text{SU}(2)$ .

Given a finite gate set  $\mathcal{G} \subseteq \text{SU}(2)$  that is closed under inverses, a *word over  $\mathcal{G}$*  is a finite product of generators and their inverses, e.g.,  $W = g_k g_{k-1} \cdots g_1$  with each  $g_j \in \mathcal{G} \cup \mathcal{G}^{-1}$ . The *length*  $|W|$  is the number of factors  $k$ , which we equate with the synthesized one-qubit gate count; equality/approximation is taken in  $\text{SU}(2)$ .

**Theorem 2.3.6** (Solovay–Kitaev). *Let  $\mathcal{G} \subseteq \text{SU}(2)$  be a finite, inverse-closed set whose generated subgroup  $\langle \mathcal{G} \rangle$  is dense in  $\text{SU}(2)$ . Then there exists a constant  $c > 0$  such that for every  $U \in \text{SU}(2)$  and every  $\varepsilon \in (0, 1)$  there is a word  $W$  over  $\mathcal{G}$  of length  $\mathcal{O}(\log^c(1/\varepsilon))$  with*

$$\|U - W\| := \sup_{|\psi\rangle : \langle\psi|\psi\rangle=1} \|(U - W)|\psi\rangle\| \leq \varepsilon, \quad (2.3.9)$$

where  $\|\cdot\|$  denotes the operator (spectral) norm. Moreover, there is a constructive classical algorithm which, given  $U$  and  $\varepsilon$ , outputs such a word in time  $\text{polylog}(1/\varepsilon)$  [40, 35].

Beyond the standard Solovay–Kitaev algorithm, number-theoretic and geometry-based synthesis methods for special gate sets achieve the optimal  $\mathcal{O}(\log(1/\varepsilon))$  scaling [43]. Nevertheless, [Theorem 2.3.6](#) remains the broadly applicable guarantee: any inverse-closed finite gate set that is dense in  $\text{SU}(2)$  supports efficient approximation of arbitrary single-qubit unitaries.

**A universal gate set.** The gates  $\{H, S, T, \text{CNOT}\}$ , where  $S = \sqrt{Z} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$  and  $T = \sqrt{S} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$ , form a universal gate set. The single-qubit gates  $\{H, S, T\}$  generate a dense subgroup of  $\text{SU}(2)$  and satisfy the conditions of [Theorem 2.3.6](#). Combined with CNOT for two-qubit entangling operations, this provides a complete toolkit for universal quantum computation [35].

**Example: Ising model time evolution.** We demonstrate how quantum circuits implement Hamiltonian simulation, a central theme of this thesis. Consider the two-qubit Ising Hamiltonian without transverse field:

$$H = -J Z_1 \otimes Z_2, \quad (2.3.10)$$

where  $J$  is the coupling strength. Time evolution under this Hamiltonian is the unitary  $U(t) = e^{iJt Z_1 \otimes Z_2}$ .  $e^{i\theta Z \otimes Z}$  can be decomposed using CNOT and single-qubit rotations:

$$e^{i\theta Z_1 \otimes Z_2} = \text{CNOT}_{1 \rightarrow 2} \cdot (I \otimes R_Z(-2\theta)) \cdot \text{CNOT}_{1 \rightarrow 2}. \quad (2.3.11)$$

This works because CNOT transforms  $Z_1 \otimes Z_2$  into  $Z_1 \otimes I$ : conjugating by CNOT converts the two-body interaction into a single-qubit rotation, which can then be implemented directly.

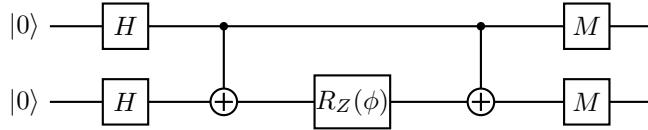


Figure 2.1: Circuit for Ising model time evolution. Hadamard gates prepare the initial state  $|++\rangle$ . The CNOT– $R_Z(\phi)$ –CNOT sequence implements the ZZ interaction  $e^{i\theta Z_1 \otimes Z_2}$  with  $\phi = -2\theta = -2Jt$ . Computational basis measurements complete the simulation.

Starting from  $|00\rangle$  and applying  $H \otimes H$  yields the product state  $|++\rangle$ . The ZZ evolution with  $\theta = Jt$  then entangles the qubits; for  $\theta = \pi/4$ , the state becomes maximally entangled. Measuring in the computational basis yields outcomes with probabilities that depend on  $\theta$ .

For more complex Hamiltonians with multiple interaction terms, product formulas (Trotter–Suzuki decompositions) combine such elementary evolutions, as developed in the next chapter. The Solovay–Kitaev theorem ensures that the rotation  $R_Z(\phi)$  can be efficiently approximated from a discrete gate set like  $\{H, S, T\}$ .

This chapter established the foundations for quantum computation. We began with the quantum postulates, states as vectors in Hilbert space, unitary evolution, measurement, and composite systems via tensor products. The density matrix formalism extended this framework to mixed states and subsystems of entangled systems. We then developed the circuit model, defining fundamental gates (Pauli, Hadamard, rotation, CNOT) and their action on both pure and mixed states. The Solovay–Kitaev theorem illustrated that finite gate sets can approximate arbitrary unitaries with only polylogarithmic overhead. We concluded with Ising model time evolution, demonstrating how Hamiltonian simulation reduces to decomposing  $e^{-iHt}$  into elementary gates. The next chapter develops systematic methods for this decomposition, introducing product formulas and more advanced simulation techniques.

## Chapter 3

# Hamiltonian Simulation

The previous chapter established the Schrödinger equation  $i\frac{d}{dt}|\psi\rangle = H|\psi\rangle$  and demonstrated Hamiltonian simulation for the two-qubit Ising model, where we decomposed the time evolution operator  $e^{-iHt}$  into a quantum circuit. That example was straightforward because the Hamiltonian consisted of a single term. Most physically interesting systems, however, have Hamiltonians with multiple non-commuting terms, making their simulation substantially more challenging. The one-dimensional nearest-neighbour transverse field Ising model (TFIM) is the canonical example:

$$H_{\text{TFIM}} = -J \sum_{\langle i,j \rangle} Z_i Z_j - h \sum_i X_i, \quad (3.0.1)$$

where  $J$  is the coupling strength between neighboring spins and  $h$  is the transverse field strength. The Ising interactions ( $ZZ$  terms) and the transverse field ( $X$  terms) do not commute because the Pauli matrices  $Z$  and  $X$  anticommute:  $ZX = -XZ$ . This non-commutativity introduces quantum fluctuations that drive a quantum phase transition from an ordered ferromagnetic phase to a disordered paramagnetic phase [44]. Similar non-commuting structure appears throughout physics, especially molecular Hamiltonians in quantum chemistry have non-commuting kinetic and electron-electron repulsion terms [45], the Fermi-Hubbard model has non-commuting hopping and on-site interaction terms relevant to metal-insulator transitions [46], and the Heisenberg model has non-commuting  $X_i X_j$ ,  $Y_i Y_j$ , and  $Z_i Z_j$  interactions [47]. Non-commutativity is precisely what makes these systems quantum mechanical.

This non-commutativity also makes Hamiltonian simulation computationally universal. The key insight is that quantum gates *are* time evolution: every unitary gate  $U$  can be written as  $U = e^{-iHt}$  for some Hamiltonian  $H$  and evolution time  $t$ . A quantum circuit is therefore a sequence of Hamiltonian evolutions, and any quantum algorithm can be recast as simulating a single many-body Hamiltonian. The Feynman-Kitaev construction makes this precise by encoding the history of a quantum computation, a superposition of all intermediate states, into the ground state of a local Hamiltonian [48]. Lloyd's 1996 result established that simulating local Hamiltonians is BQP-complete: any problem solvable by a quantum computer can be reduced to Hamiltonian simulation, and conversely, quantum computers can efficiently simulate any local Hamiltonian [20]. Quantum computers are natural simulators precisely because they already perform Hamiltonian simulation when computing.

Classical computers face an exponential barrier as they directly compute  $e^{-iHt}$  for an  $n$ -qubit Hamiltonian requiring  $2^n \times 2^n$  matrices, and even storing a single quantum state demands exponentially many amplitudes. Quantum computers bypass this by evolving the state directly, but they face the obstacle of non-commutativity. The Hamiltonians we wish to simulate typically decompose as  $H = \sum_j H_j$ , where each  $H_j$  can be simulated efficiently do not commute.

The core mathematical issue is that  $e^{-it(H_1+H_2)} \neq e^{-itH_1}e^{-itH_2}$  unless  $[H_1, H_2] = 0$ . To see why, consider the Taylor expansions. For the sum in the exponent:

$$e^{-it(H_1+H_2)} = I - it(H_1 + H_2) + \frac{(it)^2}{2}(H_1 + H_2)^2 + \dots \quad (3.0.2)$$

Expanding the second-order term gives  $(H_1 + H_2)^2 = H_1^2 + H_1 H_2 + H_2 H_1 + H_2^2$ . For the product of exponentials:

$$e^{-itH_1}e^{-itH_2} = \left[ I - itH_1 + \frac{(it)^2}{2}H_1^2 + \dots \right] \left[ I - itH_2 + \frac{(it)^2}{2}H_2^2 + \dots \right] \quad (3.0.3)$$

Multiplying and collecting terms to second order yields  $I - it(H_1 + H_2) + \frac{(it)^2}{2}(H_1^2 + 2H_1 H_2 + H_2^2) + \dots$ . The

difference at second order is:

$$\frac{(it)^2}{2}(H_1H_2 + H_2H_1) - \frac{(it)^2}{2}(2H_1H_2) = -\frac{t^2}{2}[H_1, H_2], \quad (3.0.4)$$

where  $[H_1, H_2] = H_1H_2 - H_2H_1$  is the commutator. The Baker–Campbell–Hausdorff formula captures this systematically [49],  $e^Ae^B = e^{A+B+\frac{1}{2}[A,B]+\frac{1}{12}[A,[A,B]]-\frac{1}{12}[B,[A,B]]+\dots}$ . This generates an infinite series of nested commutator corrections.

### 3.1 Circuit Synthesis

Although Hamiltonian dynamics unfold in continuous time, quantum circuits operate through discrete gate applications. Deutsch’s foundational work on quantum computational networks established that any unitary evolution  $e^{-iHt}$  can be efficiently decomposed into a finite sequence of elementary quantum gates [38]. This discretization transforms the problem from solving differential equations to constructing combinatorial gate sequences, placing Hamiltonian simulation squarely within the circuit model of computation.

The Solovay–Kitaev theorem (Theorem 2.3.6) completes this picture by showing that these elementary gates need not be arbitrary: any single-qubit unitary can be approximated to precision  $\varepsilon$  using only  $\mathcal{O}(\log^c(1/\varepsilon))$  gates from a fixed universal set like  $\{H, S, T, \text{CNOT}\}$ . The combination of Deutsch’s discretization and Solovay–Kitaev compilation means that continuous-time quantum dynamics can be realized as finite-depth circuits from a standard gate library, with overhead that scales only polylogarithmically in the target precision.

These foundational results guarantee that circuit synthesis is *possible*, but determining *how* to construct efficient circuits remains highly non-trivial. The challenge, as the previous section established, lies in non-commuting Hamiltonian terms: naive decomposition into individual exponentials introduces systematic error. Managing this error while minimizing circuit depth has spawned an entire algorithmic landscape. Product formulas (Trotter–Suzuki), randomized methods (qDRIFT), linear combinations of unitaries, and quantum signal processing each leverage different structural assumptions to achieve favorable scaling.

Representing  $H$  as a full  $2^n \times 2^n$  matrix is exponentially expensive and largely redundant. Physical Hamiltonians possess structure (locality, sparsity, symmetry) that more compact representations can exploit. How we access and represent  $H$  fundamentally shapes which algorithms apply and what resources they require. The input models below formalize these access patterns, each exposing different primitives that algorithms can leverage.

**Definition 3.1.1** (Hamiltonian). *The Hamiltonian  $H$  of an  $n$ -qubit system is the self-adjoint operator on the system Hilbert space that generates time evolution via the Schrödinger equation. In finite-dimensional models considered here,  $H$  is a Hermitian  $2^n \times 2^n$  matrix, fully specifying the closed-system dynamics.*

For finite-dimensional systems, Hermiticity  $H = H^\dagger$  ensures that  $U(t)$  is unitary for all  $t$ . Norms such as  $\|H\|$  together with structural features (locality, sparsity, available block encodings) largely determine algorithmic cost. We discuss a few representations that enable efficient simulations.

**Pauli decomposition.** Any Hermitian operator on  $n$  qubits can be written as a linear combination of Pauli strings:

$$H = \sum_{j=1}^L c_j P_j, \quad P_j \in \{I, X, Y, Z\}^{\otimes n}, \quad (3.1.1)$$

where the coefficients  $c_j$  are real. A general  $2^n \times 2^n$  Hermitian matrix requires up to  $4^n$  terms, but physical Hamiltonians typically have  $L = \text{poly}(n)$  terms due to locality or sparsity constraints. This decomposition is natural for product formula methods [20], since Pauli operators square to identity ( $P_j^2 = I$ ), each term exponentiates analytically as  $e^{-ic_j P_j t} = \cos(c_j t)I - i \sin(c_j t)P_j$ , implementable with standard gates. Algorithms in this paradigm simulate  $e^{-iHt}$  by sequentially exponentiating individual terms, managing the error from non-commutativity through Trotter–Suzuki formulas or randomized sampling.

**Linear combinations of unitaries.** When individual Hamiltonian terms do not exponentiate as conveniently as Pauli operators, a different strategy applies. Rather than exponentiating terms separately, we express the *entire* evolution operator  $e^{-iHt}$  as a linear combination of unitaries (LCU) [28]. The truncated Taylor series method realizes this directly [50] by expanding  $e^{-iHt} = \sum_{k=0}^{\infty} (-iHt)^k/k!$  and truncating at order  $K$  yielding:

$$e^{-iHt} \approx \sum_{k=0}^K \frac{(-it)^k}{k!} H^k. \quad (3.1.2)$$

If the Hamiltonian itself can be decomposed as  $H = \sum_j \alpha_j U_j$ , powers  $H^k$  expand into sums of unitary products, making the truncated series itself an LCU. This representation achieves precision scaling logarithmically in  $1/\varepsilon$ , an exponential improvement over product formulas, and extends simulation to Hamiltonians whose individual terms lack convenient exponential forms but can be expressed as a linear combination of unitaries.

**Block encoding and quantum signal processing.** More sophisticated methods bypass the Hamiltonian’s term-by-term structure entirely. These approaches construct a *block encoding*, a unitary  $W$  such that  $(\langle 0^a | \otimes I)W(|0^a\rangle \otimes I) = H/\alpha$ , embedding  $H$  into a larger unitary’s upper-left block. This new representation of the Hamiltonian permits a different simulation paradigm [10]. Given a block encoding of  $H$ , the qubitization procedure constructs an operator  $W$  whose eigenvalues encode those of  $H$  in a structured way, each eigenvalue  $\lambda$  of  $H/\alpha$  appears in a block with entries determined by  $\lambda$  and  $\sqrt{1 - \lambda^2}$ . Quantum signal processing then applies polynomial transformations to these eigenvalues [11], enabling direct implementation of  $\lambda \mapsto e^{-it\lambda}$ . The result is an algorithm that simulates  $e^{-iHt}$  holistically rather than term-by-term, achieving optimal query complexity of  $\mathcal{O}(\alpha t + \log(1/\varepsilon))$  to the block encoding oracles.

These paradigms offer different trade-offs. Product formulas are conceptually simpler, require no ancilla overhead for the decomposition itself, and their error depends on commutator structure rather than the number of terms. Block encoding methods achieve superior asymptotic scaling but require complex circuits and tricky control structures. For near-term and early fault-tolerant devices with limited ancilla counts, product formulas and randomized methods often prove more practical. For asymptotically large simulations, block encoding approaches dominate.

The following sections develop the first two paradigms in detail, product formulas that exploit Pauli decompositions, and LCU-based methods including truncated Taylor series. These approaches remain the most viable candidates for early fault-tolerant quantum computers, where ancilla overhead and circuit complexity impose practical constraints. Block encoding and quantum signal processing, while asymptotically optimal, require resources beyond current architectural capabilities and are not treated further here.

## 3.2 Pauli Decomposition Model

The foundations for quantum simulation of Hamiltonians were established by Seth Lloyd in his 1996 paper [20], supporting Feynman’s 1982 conjecture that quantum computers could efficiently simulate quantum systems. Lloyd’s result shows that local Hamiltonian simulation lies in BQP; the converse direction, that any BQP computation can be encoded as a local Hamiltonian evolution, establishes the problem’s BQP-completeness and follows from the universality of quantum circuits built from local gates. Lloyd demonstrated that for a Hamiltonian decomposed as  $H = \sum_j H_j$ , the time evolution operator  $e^{-iHt}$  can be approximated by composing short-time evolutions of individual terms using the product formula  $e^{-iHt} \approx (e^{-iH_1 t/r} e^{-iH_2 t/r} \dots e^{-iH_L t/r})^r$ . This approach, now known as Trotterization, exploits the Lie-Trotter decomposition from operator theory.

Lloyd’s analysis focused specifically on *local* Hamiltonians, those where each term  $H_j$  acts nontrivially on only a bounded number of qubits, for a fundamental practical reason. As established in the previous section, any  $n$ -qubit Hermitian operator admits a Pauli decomposition  $H = \sum_{j=1}^L c_j P_j$ , but a general Hamiltonian may require up to  $4^n$  such terms. Local Hamiltonians circumvent this exponential explosion, as a  $k$ -local Hamiltonian on  $n$  qubits contains at most  $\binom{n}{k} 4^k = \mathcal{O}(n^k)$  terms, a polynomial rather than exponential count. This locality constraint encompasses most physically relevant systems, including lattice spin models, molecular Hamiltonians with bounded interaction ranges, and condensed-matter systems with local couplings.

### 3.2.1 First-Order Trotterization

Given a decomposition  $H = \sum_{j=1}^L H_j$ , the first-order Lie-Trotter formula approximates the time evolution by composing short evolutions of each term

$$e^{-iHt} \approx \left( \prod_{j=1}^L e^{-iH_j(t/r)} \right)^r. \quad (3.2.1)$$

The error in this approximation arises from the noncommutativity of the  $H_j$  operators. As shown in the previous section, the Baker-Campbell-Hausdorff formula implies that the leading error term scales as  $\mathcal{O}(\tau^2[H_j, H_k])$ . Bounding the commutator terms by operator norms gives the *naive complexity* estimate. Writing  $\Lambda := \max_j \|H_j\|$  as the maximum term norm, we can bound each commutator using the submultiplicativity of the operator norm as  $\|[H_j, H_k]\| \leq 2\|H_j\|\|H_k\| \leq 2\Lambda^2$ . Since there are  $\binom{L}{2}$  pairs, the single-step error scales as

$\mathcal{O}(L^2\Lambda^2\tau^2)$ . After  $r$  repetitions, the total error accumulates to at most  $r \cdot \mathcal{O}(L^2\Lambda^2(t/r)^2) = \mathcal{O}(L^2\Lambda^2t^2/r)$ . To achieve target precision  $\varepsilon$ , one requires

$$r = \mathcal{O}\left(\frac{L^2\Lambda^2t^2}{\varepsilon}\right) \quad (3.2.2)$$

Trotter steps, each containing  $L$  exponentials, yielding a total gate complexity of  $\mathcal{O}(L^3\Lambda^2t^2/\varepsilon)$ .

The naive complexity (3.2.2) pessimistically bounds every commutator  $[H_j, H_k]$  by  $2\|H_j\|\|H_k\|$ , ignoring the crucial fact that many terms in physical Hamiltonians commute exactly. Consider the transverse-field Ising model introduced at the beginning of this chapter. The ZZ interaction terms on non-overlapping bonds commute with each other, since  $[Z_1Z_2, Z_3Z_4] = 0$ , and similarly all single-site  $X_i$  terms mutually commute. Only terms sharing a site produce nonzero commutators. For a general  $k$ -local Hamiltonian on  $n$  sites with geometrically local interactions, the number of non-commuting pairs scales as  $\mathcal{O}(n)$  rather than  $\mathcal{O}(L^2) = \mathcal{O}(n^2)$ , representing a substantial gap between the naive bound and actual simulation costs.

Childs, Su, Tran, Wiebe, and Zhu [22] developed a refined error analysis that replaces crude norm products with sums over actual commutator norms, capturing the structure of specific Hamiltonians. For a  $k$ -local Hamiltonian where each term acts on at most  $k$  qubits and each qubit participates in at most  $m$  terms, the first-order Trotter error satisfies

$$\left\| e^{-iH\tau} - \prod_{j=1}^L e^{-iH_j\tau} \right\| \leq \frac{\tau^2}{2} \sum_{j < \ell} \| [H_\ell, H_j] \| + \mathcal{O}(\tau^3). \quad (3.2.3)$$

Critically, each  $H_j$  overlaps with at most  $km$  other terms, specifically those sharing at least one qubit, so the double sum contains only  $\mathcal{O}(Lkm)$  nonzero contributions rather than  $\mathcal{O}(L^2)$ . For lattice systems with bounded-range interactions, this yields Trotter step counts scaling as

$$r = \mathcal{O}\left(\frac{nkm\Lambda^2t^2}{\varepsilon}\right) = \mathcal{O}\left(\frac{nt^2}{\varepsilon}\right) \quad (3.2.4)$$

for fixed  $k$ ,  $m$ , and  $\Lambda$ , which is linear in system size  $n$  rather than quadratic in the term count  $L$ .

### 3.2.2 Second-Order and Higher Formulas

The symmetric second-order Strang formula cancels leading odd-order errors by sweeping forward and backward through the terms

$$S_2(s) = \left( \prod_{j=1}^L e^{-iH_j(s/2)} \right) \left( \prod_{j=L}^1 e^{-iH_j(s/2)} \right), \quad e^{-iHt} \approx (S_2(t/r))^r. \quad (3.2.5)$$

The symmetrization eliminates the first-order error term, so the single-step error scales as  $\mathcal{O}(\tau^3)$  rather than  $\mathcal{O}(\tau^2)$ . Following the same naive bounding argument as before, the total error after  $r$  repetitions becomes  $\mathcal{O}(L^2\Lambda^2t^3/r^2)$ , requiring

$$r = \mathcal{O}\left(\sqrt{\frac{L^2\Lambda^2t^3}{\varepsilon}}\right) \quad (3.2.6)$$

Trotter steps to achieve precision  $\varepsilon$ . The improvement from  $\varepsilon^{-1}$  to  $\varepsilon^{-1/2}$  scaling motivates the search for even higher-order formulas.

Masuo Suzuki's fractal decomposition method [21] provides a systematic recursion for constructing product formulas of arbitrary even order. For a  $p$ -th order formula, the single-step error scales as  $\mathcal{O}(\tau^{p+1})$ , yielding a naive complexity of  $r = \mathcal{O}((\Lambda t)^{1+1/p}\varepsilon^{-1/p})$  Trotter steps. The precision scaling improves from  $\varepsilon^{-1}$  at first order toward  $\varepsilon^0$  as  $p \rightarrow \infty$ , but each Trotter step requires more exponentials, creating a trade-off that typically favors orders between 2 and 4 for practical implementations.

The commutator-scaling analysis of Childs et al. [22] extends to arbitrary order  $p$ . Define the adjoint map  $\text{ad}_A(B) := [A, B]$  and its iterates  $\text{ad}_A^q(B) := [A, \text{ad}_A^{q-1}(B)]$ . For a sequence of operators  $A_s, \dots, A_1, B$ , the  $p$ -th order  $\alpha$ -commutator constant is

$$\alpha_{\text{comm}}\{A_s, \dots, A_1, B\} := \sum_{\substack{q_1, \dots, q_s \geq 0 \\ q_1 + \dots + q_s = p}} \binom{p}{q_1, \dots, q_s} \left\| \text{ad}_{A_s}^{q_s} \cdots \text{ad}_{A_1}^{q_1}(B) \right\|. \quad (3.2.7)$$

For a decomposition  $H = \sum_{j=1}^L H_j$ , the  $\alpha$ -commutator envelope aggregates all depth- $p$  nested commutators

$$\tilde{\alpha}_{\text{comm}} := \sum_{j_{p+1}, \dots, j_1=1}^L \|\text{ad}_{H_{j_{p+1}}} \cdots \text{ad}_{H_{j_2}}(H_{j_1})\|. \quad (3.2.8)$$

The tilde distinguishes this aggregate envelope from the single-sequence constant  $\alpha_{\text{comm}}$  in (3.2.7). For  $p = 1$ , the envelope reduces to the pairwise commutator sum  $\tilde{\alpha}_{\text{comm}} = 2 \sum_{j < k} \| [H_j, H_k] \|$ . A  $p$ -th order product formula  $S_p(\tau)$  yields

$$\|e^{-iH\tau} - S_p(\tau)\| \leq C_p \tilde{\alpha}_{\text{comm}} \tau^{p+1}, \quad (3.2.9)$$

where  $C_p$  is a universal constant depending only on  $p$ . After  $r$  repetitions, the global error satisfies

$$\|e^{-iHt} - S_p(t/r)^r\| \leq C_p \frac{\tilde{\alpha}_{\text{comm}} t^{p+1}}{r^p}. \quad (3.2.10)$$

Meeting target precision  $\varepsilon$  requires

$$r = \mathcal{O}\left(\left(\frac{\tilde{\alpha}_{\text{comm}} t^{p+1}}{\varepsilon}\right)^{1/p}\right), \quad (3.2.11)$$

which recovers the  $\varepsilon^{-1/p}$  and  $t^{1+1/p}$  scalings while replacing coarse norm bounds with the commutator-based envelope  $\tilde{\alpha}_{\text{comm}}$ .

The commutator bounds developed in this section exploit structural properties of the Hamiltonian that go beyond merely being expressible as a Pauli decomposition. The locality and sparsity of interactions, encoded in which terms share support and thus fail to commute, determine the envelope  $\tilde{\alpha}_{\text{comm}}$  and ultimately the simulation complexity. However, commutator structure is not the only exploitable feature. Many physical Hamiltonians exhibit pronounced imbalance in the magnitudes of their coefficients: dominant terms coexist with numerous weaker contributions, as occurs in electronic structure where two-electron integrals span several orders of magnitude [45], or in long-range lattice models where coupling strengths decay with distance [51]. This coefficient heterogeneity motivates randomized approaches that weight sampling according to term importance rather than treating all summands uniformly.

### 3.2.3 Randomized compilation: qDRIFT

The product formulas developed above apply every term in the Hamiltonian during each Trotter step, treating all summands uniformly regardless of their magnitudes. Campbell's qDRIFT algorithm [24] takes a different approach. Rather than deterministically cycling through all  $L$  terms, it randomly samples which term to evolve at each step, with probabilities proportional to term strengths. This stochastic strategy shifts the complexity from depending on the term count  $L$  to depending on the  $\ell_1$ -norm  $\lambda := \sum_j \|H_j\|$ , which can be substantially smaller when many terms have small coefficients.

Concretely, write the Hamiltonian as  $H = \sum_{j=1}^L w_j \hat{H}_j$  where each term is normalized so that  $\|\hat{H}_j\| \leq 1$  and the weight  $w_j := \|H_j\| > 0$  captures the term's magnitude. The qDRIFT channel approximates time evolution by concatenating  $M$  random single-term evolutions  $e^{-i\tau \hat{H}_j}$ , where each index  $j$  is drawn independently with probability  $p_j = w_j/\lambda$ . Intuitively, stronger terms appear more frequently in the product, while weak terms contribute proportionally less. The per-step duration  $\tau = \lambda t/M$  is chosen so that the expected evolution matches the target, and the randomness averages out errors from neglecting commutators between terms.

Campbell showed that this procedure achieves diamond-norm error at most  $\varepsilon$  using

$$M = \mathcal{O}\left(\frac{\lambda^2 t^2}{\varepsilon}\right) \quad (3.2.12)$$

random steps, each implementing a single exponential  $e^{-i\tau \hat{H}_j}$ . Crucially, this bound is independent of both  $L$  and the maximum term norm  $\Lambda = \max_j w_j$ , depending only on the total weight  $\lambda$ . For molecular Hamiltonians where two-electron integrals span several orders of magnitude, or lattice models with power-law decaying couplings, many terms contribute negligibly to  $\lambda$  despite inflating  $L$ , making qDRIFT particularly advantageous. Campbell reported speedups of  $300\times$  to  $1600\times$  over first-order Trotter for molecules such as propane and carbon dioxide at precision  $10^{-3}$ . The principal complexity scalings are summarized in Table 3.1.

Algorithm 1 gives a concrete realization of the qDRIFT procedure.

---

**Algorithm 1** qDRIFT randomized compiler for Hamiltonian simulation

**Require:** Decomposition  $H = \sum_{j=1}^L w_j \hat{H}_j$  with  $w_j > 0$ ,  $\|\hat{H}_j\| \leq 1$ , time  $t$ , target error  $\varepsilon > 0$ .

- 1:  $\lambda \leftarrow \sum_{j=1}^L w_j$ .
  - 2: Choose number of random steps  $M \leftarrow \lceil 2\lambda^2 t^2 / \varepsilon \rceil$ .
  - 3: Set per-step duration  $\tau \leftarrow (\lambda t) / M$ ; initialize  $V \leftarrow I$ .
  - 4: **for**  $k = 1, \dots, M$  **do**
  - 5:     Sample index  $j \in \{1, \dots, L\}$  with probability  $p_j = w_j / \lambda$ .
  - 6:     Set  $V \leftarrow e^{-i\tau \hat{H}_j} V$ .
  - 7: **end for**
  - 8: **return**  $V$  as an  $\varepsilon$ -approximation to  $e^{-iHt}$ .
- 

The shift from term counts to weight distributions opens the door to hybrid strategies that combine randomized and deterministic elements. Ouyang, White, and Campbell [52] introduced stochastic Hamiltonian sparsification, which interpolates between qDRIFT and randomized Trotter by deterministically evolving the strongest terms while stochastically sampling weaker ones, achieving quadratic error suppression over purely deterministic sparsification. Hagan and Wiebe [53] formalized this intuition into a composite simulation framework that partitions the Hamiltonian so that large-magnitude terms are handled by high-order product formulas while numerous small terms are relegated to qDRIFT, with rigorous bounds on the combined channel’s diamond-norm error. Poernic et al. [54] extended these composite methods to imaginary-time evolution and demonstrated up to 20-fold speedups for Jellium simulations. These developments illustrate a broader theme: no single simulation primitive dominates across all Hamiltonians, and the most efficient compilation strategies exploit problem-specific structure by mixing deterministic and randomized techniques.

### 3.2.4 Recent Developments: THRIFT in the Interaction Picture

The algorithms discussed so far treat all Hamiltonian terms on equal footing, whether through deterministic Trotter sweeps or importance-weighted qDRIFT sampling. Yet many physical systems exhibit a natural hierarchy of energy scales: strong local interactions coexist with weak perturbations, as occurs when external fields probe a material or when long-range couplings decay with distance. The THRIFT framework (Time-dependent Hamiltonian simulation via Random sampling with Interaction-picture Formula and Time-ordering) exploits this structure by separating the Hamiltonian into a dominant “easy” part and a perturbative remainder, then simulating in the interaction picture defined by the dominant terms [55].

The conceptual foundation traces to Low and Wiebe’s 2018 observation that interaction-picture methods can achieve complexity scaling with the perturbation strength  $\alpha$  rather than the full Hamiltonian norm [56]. Their approach used truncated Dyson series and linear combinations of unitaries, requiring ancilla qubits, block encodings, and controlled reflections. The same interaction-picture transformation could instead be combined with product formulas, eliminating ancillas entirely while retaining the favorable  $\alpha$ -dependence. This insight yields a method well-suited to near-term devices: no auxiliary qubits, no multi-controlled gates, and circuit structures that match hardware-native gate sets.

**First-order THRIFT decomposition.** Consider  $H = H_0 + \alpha H_1$  where  $H_0$  generates dynamics we can implement cheaply (perhaps it is diagonal or consists of commuting terms) and  $\alpha H_1$  is a weak perturbation. Standard Trotter would alternate between  $e^{-iH_0\tau}$  and  $e^{-i\alpha H_1\tau}$ , incurring error from their failure to commute. THRIFT instead performs a change of reference frame for the entire problem. Define the interaction-picture state  $|\tilde{\psi}(t)\rangle = e^{iH_0 t} |\psi(t)\rangle$ . Taking the time derivative and substituting the Schrödinger equation  $i\partial_t |\psi\rangle = (H_0 + \alpha H_1) |\psi\rangle$  yields

$$i \frac{d}{dt} |\tilde{\psi}\rangle = iH_0 e^{iH_0 t} |\psi\rangle + e^{iH_0 t} (-i(H_0 + \alpha H_1)) |\psi\rangle = \alpha e^{iH_0 t} H_1 e^{-iH_0 t} |\tilde{\psi}\rangle \equiv \alpha \tilde{H}_1(t) |\tilde{\psi}\rangle. \quad (3.2.13)$$

The  $H_0$  contributions cancel exactly, leaving only the rotated perturbation  $\tilde{H}_1(t) = e^{iH_0 t} H_1 e^{-iH_0 t}$ . After evolving  $|\tilde{\psi}(t)\rangle$  under this time-dependent Hamiltonian, one recovers the original state via  $|\psi(t)\rangle = e^{-iH_0 t} |\tilde{\psi}(t)\rangle$ . The  $H_0$  evolution is handled exactly by the frame transformation; only the perturbation requires approximation.

If the perturbation decomposes as  $H_1 = \sum_\gamma H_{1\gamma}$ , the first-order THRIFT approximation takes the form

$$U_{\text{apx}}(t) := e^{-itH_0} \prod_\gamma e^{itH_0} e^{-it(H_0 + \alpha H_{1\gamma})}. \quad (3.2.14)$$

Each factor  $e^{itH_0}e^{-it(H_0+\alpha H_{1\gamma})}$  evolves under  $H_0 + \alpha H_{1\gamma}$  and then undoes the  $H_0$  part, isolating the contribution of  $H_{1\gamma}$  in the interaction frame. Because errors from noncommutativity with  $H_0$  are absorbed into the exact frame transformation, only commutators among the weak terms themselves contribute to the leading error. The approximation error scales as  $\mathcal{O}(\alpha^2 t^2)$ , improving upon standard first-order Trotter's  $\mathcal{O}(\alpha t^2)$  by a factor of  $\alpha$ . Remarkably, numerical experiments show that THRIFFT remains competitive even when  $\alpha$  is not particularly small, suggesting that the method captures favorable commutator structure beyond the strictly perturbative regime.

THRIFFT is especially advantageous when: (i)  $\alpha \ll 1$  (perturbative regime), (ii)  $H_0$  is easy to implement exactly (e.g., diagonal single-qubit terms), and (iii) commutators among weak terms are small (local or near-commuting structure). Under these conditions, first-order THRIFFT lowers the number of steps for a fixed error budget by a factor  $\sim \alpha^2$  relative to comparable Trotter sequences. For the one-dimensional transverse-field Ising model  $H = -J \sum_i Z_i Z_{i+1} - h \sum_i X_i$ , taking  $H_0$  as the ZZ interactions and the transverse field as the perturbation, Santos et al. report that THRIFFT matches standard Trotter in per-step gate count (both require  $2N$  two-qubit gates per step for an  $N$ -site chain) while achieving order-of-magnitude improvements in simulated system size and evolution time within a fixed gate budget.

### 3.3 LCU Decomposition Model

Quantum gates implement unitary transformations, yet many quantum algorithms require applying non-unitary operators that cannot be directly executed on quantum circuits. The *Linear Combination of Unitaries* (LCU) framework provides a systematic procedure to implement such non-unitary operators via unitary quantum circuits augmented with measurement. Efficient LCU-based algorithms therefore exploit problem-specific structure such as sparsity, locality, or symmetry to obtain compact decompositions with far fewer terms. In the context of Hamiltonian simulation, truncated Taylor series methods [57] and more advanced approaches such as qubitization and quantum signal processing [10, 58] exploit this structure to achieve near-optimal query complexity.

Any linear operator  $A$  can be approximated as a weighted sum of unitary operators,

$$\tilde{A} = \sum_{j=1}^M \alpha_j U_j, \quad \alpha_j \in \mathbb{R}_+, \quad U_j^\dagger U_j = I,$$

where the coefficients  $\alpha_j$  are taken to be real and positive without loss of generality, since any complex phase can be absorbed into the corresponding unitary  $U_j$ . The approximation satisfies  $\|\tilde{A} - A\| \leq \gamma$  for a target error  $\gamma$ . Given an LCU decomposition, the framework implements  $\tilde{A}$  by encoding the coefficients into an ancilla register prepared in superposition. Controlled operations then coherently apply each unitary  $U_j$  weighted by the corresponding amplitude, and a final measurement of the ancilla projects onto the desired linear combination. The standard LCU algorithm employs a three-stage *prepare-select-unprepare* procedure, summarized in Algorithm 2.

---

**Algorithm 2** Standard LCU (Prepare-Select-Unprepare)

**Require:** LCU decomposition  $\tilde{A} = \sum_{j=1}^M \alpha_j U_j$  with  $\alpha := \sum_{j=1}^M \alpha_j$   
**Require:** Input state  $|\psi\rangle$ , ancilla register of  $m = \lceil \log_2 M \rceil$  qubits  
**Ensure:** Output state proportional to  $\tilde{A} |\psi\rangle$  upon successful postselection

- 1: **Prepare:** Apply unitary  $R$  such that  $R |\bar{0}\rangle = \frac{1}{\sqrt{\alpha}} \sum_{j=1}^M \sqrt{\alpha_j} |j\rangle$
- 2: **Select:** Apply controlled-unitary  $Q = \sum_{j=1}^M |j\rangle\langle j| \otimes U_j$
- 3: **Unprepare:** Apply  $R^\dagger$  to the ancilla register
- 4: **Measure:** Measure ancilla in computational basis
- 5: **if** outcome is  $|\bar{0}\rangle$  **then**
- 6:     **return** system register contains  $\tilde{A} |\psi\rangle / \alpha$
- 7: **else**
- 8:     **return** failure
- 9: **end if**

---

The complete transformation induced by the circuit is

$$(R^\dagger \otimes I) Q (R \otimes I) |\bar{0}\rangle^{\otimes m} |\psi\rangle = \frac{1}{\alpha} \sum_{j=1}^M \alpha_j U_j |\psi\rangle \otimes |\bar{0}\rangle + |\Phi_\perp\rangle,$$

where  $|\Phi_{\perp}\rangle$  has the ancilla orthogonal to  $|\bar{0}\rangle$ . Postselecting on the ancilla measurement outcome  $|\bar{0}\rangle$  yields the desired state  $\tilde{A}|\psi\rangle/\alpha$  in the system register. When the success probability is low, amplitude amplification can boost it to near unity [57].

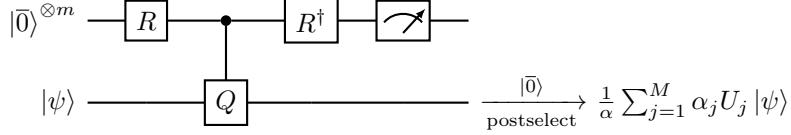


Figure 3.1: Standard LCU circuit implementing the prepare-select-unprepare protocol

While asymptotically efficient, the standard LCU construction imposes substantial resource demands. The protocol requires  $m = \lceil \log M \rceil$  ancilla qubits, and the select oracle involves multi-controlled unitaries where all  $m$  ancilla qubits jointly control each  $U_j$ , resulting in deep circuits with significant non-Clifford gate overhead. The success probability scales as  $\sum_j \alpha_j^2 / \alpha^2$ , potentially as low as  $1/M$  in the worst case, necessitating amplitude amplification. Finally, state preparation for the prepare unitary  $R$  can require exponentially many gates for arbitrary weight distributions. These constraints make standard LCU viable only with full-scale fault-tolerant quantum computation.

Beyond Hamiltonian simulation, LCU underpins advanced quantum algorithms including qubitization and quantum singular value transformation [10, 58], which use LCU-based block encodings to implement general matrix functions, and ground state preparation methods that filter eigenstates through LCU-implemented projectors [10]. The resource demands of standard LCU propagate to all dependent algorithms, making efficient LCU implementations particularly impactful. For many applications, explicit state preparation is unnecessary; instead, one seeks to estimate expectation values  $\text{Tr}[O\rho]$  for observables  $O$ . This motivates an alternative implementation that bypasses the resource demands of the standard protocol. By reformulating the problem as one of randomized sampling rather than coherent superposition, the single-ancilla LCU method reduces the ancilla requirement to a single qubit, eliminates multi-controlled gates entirely, and trades coherent circuit depth for classical repetitions. This approach is particularly well-suited to early fault-tolerant devices where qubit count and gate fidelity are the primary bottlenecks.

### 3.4 Single–Ancilla LCU Framework

This section develops a pragmatic variant of LCU designed for early fault-tolerant devices with very limited spare ancilla. The core idea is to replace full multi-ancilla LCU state preparation and deterministic coherent selection with a single-ancilla protocol plus classical repetition to estimate expectation values of interest, while retaining the favorable precision scaling inherited from truncated-Taylor/LCU methods [57, 10, 9]. Polylogarithmic depth with minimal ancilla overhead and no complex control infrastructure makes the method well-suited to early fault-tolerant devices, where shorter circuits accumulate less noise per run and are more amenable to error mitigation techniques. The single-ancilla approach achieves polylogarithmic per-run dependence on  $1/\varepsilon$  using only one ancilla and no multi-qubit controlled gates.

The method works by exploiting linearity of expectation: rather than coherently superposing all LCU terms, we sample pairs of unitaries  $(V_1, V_2)$  from the distribution  $p_j = |c_j|/\|c\|_1$  and average their contributions classically. Each circuit applies  $V_1$  controlled on the ancilla state  $|1\rangle$  and  $V_2$  anti-controlled on  $|0\rangle$ , creating interference between the two terms. Measuring  $X \otimes O$  extracts a random variable whose expectation equals the desired observable. The  $\|c\|_1^2$  prefactor in the estimator corrects for importance sampling. We sample proportionally to  $|c_j|$  but the true weights are  $c_j$  themselves.

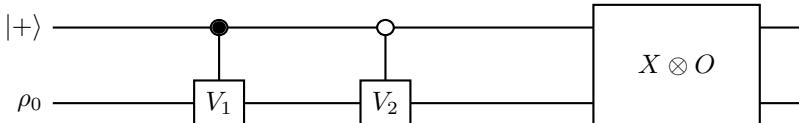


Figure 3.2: Single-ancilla LCU circuit.  $V_1$  is applied controlled on  $|1\rangle$ ,  $V_2$  anti-controlled on  $|0\rangle$ . Measuring  $X \otimes O$  yields a random outcome whose average converges to  $\text{Tr}[O f(H)\rho_0 f(H)^\dagger]$ .

**Estimator.** For a unitary  $f(H)$  (e.g.,  $e^{-iHt}$ ), the sample mean  $\hat{\mu} = \frac{\|c\|_1^2}{T} \sum_{r=1}^T \mu_r$  concentrates to  $\text{Tr}[O f(H)\rho_0 f(H)^\dagger]$ . For non-unitary  $f(H)$ , run once with  $O = I$  to estimate the normalization and output the ratio (accuracy conditions stated below).

**Algorithm 3** Single-Ancilla LCU Expectation Value Estimation

---

**Require:** LCU decomposition  $f(H) \approx \sum_j c_j U_j$  with  $\|c\|_1 = \sum_j |c_j|$

**Require:** Input state  $\rho_0$ , observable  $O$ , number of samples  $T$

**Ensure:** Estimate  $\hat{\mu}$  of  $\text{Tr}[O f(H)\rho_0 f(H)^\dagger]$

- 1: Define distribution  $\mathcal{D} \sim \{c_j/\|c\|_1, U_j\}$
- 2: **for**  $r = 1$  to  $T$  **do**
- 3:     Prepare  $|+\rangle\langle+| \otimes \rho_0$
- 4:     Sample  $V_1, V_2 \sim \mathcal{D}$
- 5:     Apply single-ancilla circuit with  $V_1, V_2$
- 6:     Measure  $X \otimes O$  and record outcome  $\mu_r$
- 7: **end for**
- 8:  $\hat{\mu} \leftarrow \|c\|_1^2 \frac{1}{T} \sum_{r=1}^T \mu_r$
- 9: **if**  $f(H)$  is non-unitary **then**
- 10:     Estimate normalization separately with  $O = I$  and divide
- 11: **end if**
- 12: **return**  $\hat{\mu}$

---

**Complexity (general).** Let  $\tau_{\rho_0}$  be the cost to prepare  $\rho_0$ , and let  $\tau_j$  be the cost to implement  $U_j$ . One coherent run has average depth  $2\langle\tau\rangle + \tau_{\rho_0}$  where  $\langle\tau\rangle = \sum_j (c_j/\|c\|_1)\tau_j$ . To achieve additive error  $\varepsilon$  with failure probability  $\delta$ ,

$$T = \mathcal{O}\left(\frac{\|O\|^2 \ln(1/\delta)}{\varepsilon^2}\right),$$

whenever the chosen LCU approximation error is set according to the robustness bounds stated below. Ancilla cost is exactly one qubit; no multi-qubit controls are required [9].

### 3.4.1 Specialization to Hamiltonian simulation

In many physical applications of Hamiltonian simulation, the goal is not to prepare the time-evolved state  $|\psi(t)\rangle = e^{-iHt}|\psi_0\rangle$  explicitly, but rather to estimate expectation values of observables:

$$\langle O \rangle_t = \text{Tr}[O \rho_t] = \text{Tr}[O e^{-iHt} \rho_0 e^{iHt}].$$

This includes energy expectation values and time-dependent correlation functions [20, 59]. This observation makes Hamiltonian simulation a natural application of the single-ancilla LCU framework, which is designed precisely for estimating quantities of the form  $\text{Tr}[O f(H)\rho_0 f(H)^\dagger]$ . A Hamiltonian expressed in the Pauli basis,  $H = \sum_{k=1}^L c_k P_k$  where each  $P_k \in \{I, X, Y, Z\}^{\otimes n}$ , immediately provides an LCU structure as every Pauli operator is unitary ( $P_k^\dagger P_k = I$ ). The normalized representation

$$\tilde{H} = H/\lambda = \sum_{k=1}^L p_k P_k, \quad p_k = |c_k|/\lambda, \quad \sum_k p_k = 1, \tag{3.4.1}$$

the coefficients  $\{p_k\}$  define a probability distribution from which Pauli operators can be efficiently sampled. This structure simplifies both the implementation and the analysis, since the normalized time  $\tilde{t} = \lambda t$  governs the algorithm's complexity and sampling occurs with the probability distribution  $\{p_k\}$ .

A direct Taylor expansion of  $e^{-iHt}$  for the full evolution time is impractical as the Taylor series approximate functions well near a reference point, and accuracy degrades as you move farther away. For the matrix exponential  $e^{-i\tau H}$ , the “distance” from the reference point is measured by  $|\tau| \cdot \|H\|$ . When this product is large, the series requires many terms to converge, and the coefficients in the resulting LCU decomposition grow correspondingly large. Since single-ancilla LCU sample complexity scales with the fourth power of the coefficient norm  $\|c\|_1^4$  (Theorem 3.4.1), even modest coefficient growth renders the algorithm impractical. Expanding  $e^{-i\tilde{H}\tilde{t}}$  where  $\tilde{t} = \lambda t$  yields an LCU whose coefficient sum grows exponentially [50, 9]:

$$\sum_j |\alpha_j| = \mathcal{O}(e^{\tilde{t}^2}) = \mathcal{O}(e^{\lambda^2 t^2}).$$

For even modest  $\lambda t = 10$ , this gives  $\|c\|_1 \sim e^{100}$ , making the  $\|c\|_1^4$  factor in the sample complexity very large.

The solution is to *segment* the evolution into  $r$  smaller pieces, each with a short enough duration that its Taylor series converges rapidly with bounded coefficients:

$$e^{-iHt} = \left( e^{-iHt/r} \right)^r = \left( e^{-i\tilde{H}\tilde{t}/r} \right)^r.$$

Each segment  $S_r = e^{-i\tilde{H}\tilde{t}/r}$  now has coefficient sum  $\sum_m |\alpha_m| \leq e^{\tilde{t}^2/r^2}$ . When we compose  $r$  such segments, the global LCU

$$S = S_r^r = \sum_{m_1, \dots, m_r} \alpha_{m_1} \cdots \alpha_{m_r} U_{m_1} \cdots U_{m_r} = \sum_j c_j W_j$$

has coefficient sum

$$\|c\|_1 = \left( \sum_m |\alpha_m| \right)^r \leq e^{\tilde{t}^2/r}.$$

Choosing  $r = \tilde{t}^2 = \lambda^2 t^2$  yields  $\|c\|_1 \leq e^1 = \mathcal{O}(1)$ —a bounded constant independent of the evolution time. This choice also ensures each segment’s evolution parameter  $\tilde{t}/r = 1/\lambda t < 1$ , guaranteeing rapid convergence of the per-segment Taylor series. With the segmentation parameter established, we now construct the explicit LCU decomposition for each segment.

Truncating  $S_r = e^{-i\tilde{H}\tilde{t}/r}$  to  $K$  terms yields

$$\tilde{S}_r = \sum_{k=0}^K \frac{(-i\tilde{H}/r)^k}{k!}.$$

By choosing

$$K = \mathcal{O} \left( \frac{\log(r/\gamma)}{\log \log(r/\gamma)} \right),$$

we ensure that  $\|S_r - \tilde{S}_r\| \leq \gamma/r$ .

To construct an explicit LCU decomposition of  $\tilde{S}_r$ , we follow Wan et al. [60] and write

$$\tilde{S}_r = \sum_{k=0}^K \frac{(-i\tilde{H}/r)^k}{k!} \quad (3.4.2)$$

$$= \sum_{k=0, k \in \text{even}}^K \frac{1}{k!} (-i\tilde{H}/r)^k \left( I - \frac{i\tilde{H}/r}{k+1} \right) \quad (3.4.3)$$

$$= \sum_{k=0, k \in \text{even}}^K \frac{1}{k!} \left( -it/r \sum_{\ell=1}^L p_\ell P_\ell \right)^k \left( I - \frac{it/r}{k+1} \left( \sum_{m=1}^L p_m P_m \right) \right) \quad (3.4.4)$$

$$= \sum_{k=0, k \in \text{even}}^K \frac{(-it/r)^k}{k!} \sum_{\ell_1, \ell_2, \dots, \ell_k=1}^L p_{\ell_1} p_{\ell_2} \cdots p_{\ell_k} P_{\ell_1} P_{\ell_2} \cdots P_{\ell_k} \sum_{m=1}^L p_m \left( I - \frac{itP_m/r}{k+1} \right) \quad (3.4.5)$$

$$= \sum_{k=0, k \in \text{even}}^K \frac{(-it/r)^k}{k!} \sqrt{1 + \left( \frac{it/r}{k+1} \right)^2} \sum_{\ell_1, \ell_2, \dots, \ell_k, m=1}^L p_{\ell_1} p_{\ell_2} \cdots p_{\ell_k} p_m P_{\ell_1} P_{\ell_2} \cdots P_{\ell_k} e^{-i\theta_m P_m}, \quad (3.4.6)$$

where  $e^{-i\theta_m P_m}$  is a Pauli rotation operator defined by

$$e^{i\theta_m P_m} = \frac{1}{\sqrt{1 + \left( \frac{it/r}{k+1} \right)^2}} \left( I - \frac{itP_m/r}{k+1} \right), \quad (3.4.7)$$

with rotation angle

$$\theta_m = \arccos \left( \left[ 1 + \left( \frac{it/r}{k+1} \right)^2 \right]^{-1/2} \right). \quad (3.4.8)$$

Thus,  $\tilde{S}_r = \sum_{j \in M} \alpha_j U_j$ , where the index set is

$$M = \{(k, \ell_1, \ell_2, \dots, \ell_k, m) : 0 \leq k \leq K; \ell_1, \ell_2, \dots, \ell_k, m \in \{1, 2, \dots, L\}\}.$$

The LCU coefficient for index  $j = (k, \ell_1, \dots, \ell_k, m)$  is

$$\alpha_j = \frac{(\tilde{t}/r)^k}{k!} \sqrt{1 + \left(\frac{\tilde{t}/r}{k+1}\right)^2} p_{\ell_1} p_{\ell_2} \cdots p_{\ell_k} p_m, \quad (3.4.9)$$

and the unitary is

$$U_j = (-i)^k P_{\ell_1} P_{\ell_2} \cdots P_{\ell_k} e^{i\theta_m P_m}. \quad (3.4.10)$$

The sum of LCU coefficients satisfies

$$\sum_{j \in M} |\alpha_j| = \sum_{k=0, k \in \text{even}}^K \frac{(\tilde{t}/r)^k}{k!} \sqrt{1 + \left(\frac{\tilde{t}/r}{k+1}\right)^2} \sum_{\ell_1, \ell_2, \dots, \ell_k, m=1}^L p_{\ell_1} p_{\ell_2} \cdots p_{\ell_k} p_m \quad (3.4.11)$$

$$= \sum_{k=0, k \in \text{even}}^K \frac{(\tilde{t}/r)^k}{k!} \sqrt{1 + \left(\frac{\tilde{t}/r}{k+1}\right)^2} \quad (3.4.12)$$

$$\leq \sum_{k=0, k \in \text{even}}^{\infty} \frac{(\tilde{t}/r)^k}{k!} \sqrt{1 + \left(\frac{\tilde{t}/r}{k+1}\right)^2} = \sum_{k=0}^{\infty} \frac{(\tilde{t}/r)^{2k}}{(2k)!} \sqrt{1 + \left(\frac{\tilde{t}/r}{2k+1}\right)^2} \quad (3.4.13)$$

$$\leq \sum_{k=0}^{\infty} \frac{(\tilde{t}/r)^{2k}}{k!} = e^{\tilde{t}^2/r^2}. \quad (3.4.14)$$

Finally, the global LCU is obtained by raising  $\tilde{S}_r$  to the  $r$ th power:

$$S = \left( \sum_{j \in M} \alpha_j U_j \right)^r = \sum_{j_1, j_2, \dots, j_r \in M} \alpha_1 \alpha_2 \cdots \alpha_r U_{j_1} U_{j_2} \cdots U_{j_r} = \sum_m c_m W_m, \quad (3.4.15)$$

where  $\|c\|_1 = \sum_m |c_m| = (\sum_{j \in M} |\alpha_j|)^r \leq e^{\tilde{t}^2/r}$ . Choosing

$$r = \tilde{t}^2 = \lambda^2 t^2 \quad (3.4.16)$$

ensures  $\|c\|_1 = \mathcal{O}(1)$ . Moreover, for this choice of  $r$  and

$$\gamma \leq \frac{\varepsilon}{6\|O\|},$$

truncating the Taylor series at

$$K = \mathcal{O} \left( \frac{\log(\lambda t \|O\|/\varepsilon)}{\log \log(\lambda t \|O\|/\varepsilon)} \right) \quad (3.4.17)$$

guarantees

$$\|e^{-itH} - S\| \leq \frac{\varepsilon}{6\|O\|}.$$

### Sampling $V_1$ and $V_2$

To apply Algorithm 3, we must sample  $V_1$  and  $V_2$  such that  $\mathbb{E}[V_1] = \mathbb{E}[V_2] = S/\|c\|_1$ . The sampling strategy proceeds as follows.

First, select an even integer  $k \in [0, K]$  according to the distribution induced by the coefficients  $\alpha_j / \sum_j \alpha_j$  from Eq. (3.4.9). Next, sample  $k+1$  Pauli operators:  $P_{\ell_1}, P_{\ell_2}, \dots, P_{\ell_k}$  and  $P_m$ , where each  $P_{\ell_i}$  is drawn independently from the probability distribution  $\{p_{\ell_i}\}_{\ell_i=1}^L$ , and  $P_m$  is sampled from  $\{p_m\}_{m=1}^L$ . From this sampling, construct the product unitary

$$W_1 = (-i)^k P_{\ell_1} P_{\ell_2} \cdots P_{\ell_k} e^{i\theta_m P_m},$$

consisting of  $k$  Pauli operators and a single Pauli rotation. Repeat this procedure  $r$  times to obtain unitaries  $W_1, W_2, \dots, W_r$ , and form the composite

$$W = W_r W_{r-1} \cdots W_1.$$

This sampling procedure produces a unitary  $W$  with  $\mathbb{E}[W] = S/\|c\|_1$ , which allows direct application of Algorithm 3.

### Running Time and Formal Complexity Theorem

The quantum circuit for Algorithm 3 implements controlled (anti-controlled) versions of  $V_1$  ( $V_2$ ). To estimate the cost of each run, we analyze the circuit depth of  $V_1$  and  $V_2$ . From the sampling procedure in §3.4.1, each sampled unitary is a product of  $r$  unitaries  $W = W_r \cdots W_1$ , where each  $W_j$  is itself a product of at most  $K + 1$  unitaries— $K$  Pauli operators and one Pauli rotation. Thus the circuit depth of  $V_1$  is at most  $(K + 1)r$ , and the overall circuit depth for both  $V_1$  and  $V_2$  is  $2(K + 1)r = \mathcal{O}(Kr)$ . Substituting the choices of  $r$  and  $K$  from Eqs. (3.4.16) and (3.4.17), the cost of each coherent run is  $2\tau_{\max} + \tau_{\rho_0}$ , where

$$\tau_{\max} = \mathcal{O}(Kr) = \mathcal{O}\left(\lambda^2 t^2 \frac{\log(\lambda t \|O\|/\varepsilon)}{\log \log(\lambda t \|O\|/\varepsilon)}\right). \quad (3.4.18)$$

We require the following theorem from Chakraborty [9] to bound the total number of repetitions:

**Theorem 3.4.1** (Estimating expectation values of observables). *Let  $\varepsilon, \delta, \gamma \in (0, 1)$  be parameters. Let  $O$  be an observable and  $\rho_0$  be an initial state. Suppose there is a Hermitian matrix  $H \in \mathbb{R}^{N \times N}$ , such that  $\|f(H) - \sum_j c_j U_j\| \leq \gamma$ , where  $U_j$  are unitaries and*

$$\gamma \leq \frac{\varepsilon}{6\|O\|\|f(H)\|}.$$

If

$$T \geq \frac{8\|O\|^2 \ln(2/\delta) \|c\|_1^4}{\varepsilon^2},$$

then Algorithm 3 estimates  $\mu$  such that

$$|\mu - \text{Tr}[O f(H) \rho_0 f(H)^\dagger]| \leq \varepsilon,$$

with probability at least  $1 - \delta$ , using one ancilla qubit and  $T$  repetitions of the single-ancilla circuit.

The total number of repetitions follows from Theorem 3.4.1. For the choice  $r = \lambda^2 t^2$ , we have  $\|c\|_1 = \mathcal{O}(1)$ . Therefore,

$$T = \mathcal{O}\left(\frac{\|O\|^2 \ln(2/\delta)}{\varepsilon^2}\right) \quad (3.4.19)$$

repetitions ensure that Algorithm 3 outputs  $\mu$  satisfying  $|\mu - \text{Tr}[O e^{-iHt} \rho_0 e^{iHt}]| \leq \varepsilon$  with probability at least  $1 - \delta$ . We formally state this result below.

**Theorem 3.4.2** (Single-Ancilla LCU for Hamiltonian Simulation). *Let  $H$  be a Hamiltonian such that  $H = \sum_{k=1}^L c_k P_k$ , where each  $P_k$  is a Pauli operator and  $c_k > 0$  with  $\lambda = \sum_k |c_k|$ . Suppose  $\rho_0$  is an initial state prepared in cost  $\tau_{\rho_0}$ , and  $O$  is an observable. Provided*

$$\gamma \leq \frac{\varepsilon}{6\|O\|},$$

such that  $\|e^{-itH} - S\| \leq \gamma$  and

$$T = \mathcal{O}\left(\frac{\|O\|^2 \ln(2/\delta)}{\varepsilon^2}\right),$$

Algorithm 3 outputs  $\mu$  such that

$$|\mu - \text{Tr}[O \rho_t]| \leq \varepsilon,$$

using  $T$  repetitions of the quantum circuit in Fig. 3.2. Each run costs at most  $2\tau_{\max} + \tau_{\rho_0}$ , where

$$\tau_{\max} = \mathcal{O}\left(\lambda^2 t^2 \frac{\log(\lambda t \|O\|/\varepsilon)}{\log \log(\lambda t \|O\|/\varepsilon)}\right).$$

*Proof.* We apply Theorem 3.4.1 (from [9]), substituting  $f(H) = e^{-itH}$  and noting  $\|f(H)\| = 1$ . For the chosen  $\gamma$ , we require

$$T = \mathcal{O}\left(\frac{\|O\|^2 \ln(1/\delta)}{\varepsilon^2}\right)$$

repetitions. The cost of each run is  $2\tau_{\max} + \tau_{\rho_0}$ , where  $\tau_{\max}$  is given by Eq. (3.4.18).  $\square$

When the observable is itself a linear combination of Pauli operators,  $O = \sum_k g_k P_k$ , the measurement can be performed by sampling: in each run, measure a randomly chosen  $P_k$  according to the distribution  $|g_k|/\alpha$  where  $\alpha = \sum_k |g_k|$  [9]. Since each Pauli operator has unit norm  $\|P_k\| = 1$ , the per-measurement cost uses  $\|O\| = 1$ . However, the random sampling of the observable itself introduces additional variance proportional to  $\alpha^2$ , so the total number of repetitions becomes  $\mathcal{O}(\alpha^2 \ln(1/\delta)/\varepsilon^2)$  rather than  $\mathcal{O}(\|O\|^2 \ln(1/\delta)/\varepsilon^2)$ .

**Complexity (simulation case).** With  $\tau_{\rho_0}$  as above, one coherent run has depth

$$\tau_{\max} = \mathcal{O}\left(\lambda^2 t^2 \frac{\log(\lambda t \|O\|/\varepsilon)}{\log \log(\lambda t \|O\|/\varepsilon)}\right),$$

and the number of samples is

$$T = \mathcal{O}\left(\frac{\|O\|^2 \ln(1/\delta)}{\varepsilon^2}\right).$$

Thus the overall gate depth is  $\tilde{\mathcal{O}}(\lambda^2 t^2 \|O\|^2/\varepsilon^2)$  (polylog factors as shown), with one ancilla and no multi-qubit controls [9].

### 3.4.2 Comparison with Other Methods

Single-ancilla LCU occupies an intermediate position between product formulas and standard multi-ancilla LCU. Compared to first-order product formulas and their randomized variants, it achieves polylogarithmic (rather than polynomial) per-run dependence on  $1/\varepsilon$  while using only one ancilla and avoiding multi-controlled gates. Standard multi-ancilla LCU methods achieve better overall complexity via coherent amplitude estimation, but require  $\log M$  ancillas for  $M$  terms, deep controlled-select circuits, and sophisticated state preparation. For early fault-tolerant devices, single-ancilla LCU trades higher repetition counts for dramatically simpler circuits.

Table 3.1 compares asymptotic complexities across methods. The Trotterization bounds use commutator-scaling theory [22], replacing crude norm products with the  $\alpha$ -commutator envelope  $\tilde{\alpha}_{\text{comm}}$  (defined in Section 3.2). The remaining notation:  $L$  is the term count,  $\alpha$  is the THRIFT perturbative parameter,  $t$  is evolution time,  $\varepsilon$  is target error, and  $O$  is the observable.

Table 3.1: Asymptotic complexity comparison. Trotterization bounds use commutator-scaling theory.

Simulation Algorithm	Gate Count	Sample Complexity
1st Order Trotterization	$\mathcal{O}\left(\frac{L\tilde{\alpha}_{\text{comm}} t^2}{\varepsilon}\right)$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
2nd Order Trotterization	$\mathcal{O}\left(\frac{L\tilde{\alpha}_{\text{comm}}^{1/2} t^{3/2}}{\varepsilon^{1/2}}\right)$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
( $2k$ )th Order Trotterization	$\mathcal{O}\left(\frac{L\tilde{\alpha}_{\text{comm}}^{1/(2k)} t^{1+1/(2k)}}{\varepsilon^{1/(2k)}}\right)$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
qDRIFT Protocol	$\mathcal{O}\left(\frac{(\lambda t)^2}{\varepsilon}\right)$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
THRIFT Protocol	$\mathcal{O}\left(\frac{\alpha^2 t^2}{\varepsilon}\right)$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
Single Ancilla LCU	$\mathcal{O}\left(\frac{\lambda^2 t^2 \log(\lambda t/\varepsilon)}{\log \log(\lambda t/\varepsilon)}\right)$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$

For structured systems, the  $\alpha$ -commutator bounds yield dramatically tighter estimates than naive norm products, often reducing effective simulation cost by orders of magnitude. The commutator envelope  $\tilde{\alpha}_{\text{comm}}$  scales as  $\mathcal{O}(n)$  for  $k$ -local Hamiltonians on  $n$  sites with bounded-range interactions, compared to the naive  $\mathcal{O}(L^2 \Lambda^2)$  bound.

The numerical benchmarks presented in Section 3.5 apply these commutator-scaling refinements to first- and second-order Trotterization for each test system, computing system-specific  $\tilde{\alpha}_{\text{comm}}$  values and thereby obtaining realistic resource estimates. This approach reveals that the optimal simulation method is system-dependent: product formulas with tight commutator bounds dominate for sparse, near-neighbor Hamiltonians at moderate precision, while methods such as qDRIFT and single-ancilla LCU become advantageous for dense Hamiltonians with many terms or when extremely high precision is required.

## 3.5 Benchmarking and Numerical Simulations

Classical simulation of quantum algorithms requires careful optimization to handle the exponential state space. Our implementation<sup>i</sup> employs several strategies to reduce computational overhead. First, memoization of inter-

<sup>i</sup><https://github.com/Simultonian/Ising-Simulation>

mediate unitary operators avoids redundant matrix construction. Converting abstract Pauli string representations to dense  $2^n \times 2^n$  matrices costs  $\mathcal{O}(4^n)$  operations; caching these conversions, along with the time-evolution unitaries  $e^{-iH_j\tau}$  and their controlled versions, eliminates repeated computation when identical operators appear across multiple samples.

The most significant optimization is deduplication of random samples. Recall from Section 3.4.1 that each term in the truncated Taylor series  $\tilde{S}_r$  is indexed by a tuple  $(k, \ell_1, \dots, \ell_k, m)$ , where  $k$  is the Taylor order and the remaining indices specify which Pauli operators to multiply. Each run of Algorithm 3 samples such indices independently for  $V_1$  and  $V_2$ . Because the coefficient distribution is highly non-uniform—lower-order terms dominate—identical index tuples appear with high probability across the  $N$  repetitions. Rather than computing the quantum state evolution for each sample independently, we group identical tuples, compute once per unique configuration, and weight results by multiplicity. This reduces the number of expensive state evolution calculations from  $N$  to the number of unique samples, typically orders of magnitude smaller.

By restricting to pure initial states  $|\psi\rangle$  rather than mixed states  $\rho$ , we perform state vector simulation instead of density matrix simulation. Each Trotter step or LCU application reduces to a matrix-vector multiplication  $U|\psi\rangle$  costing  $\mathcal{O}(4^n)$  operations, rather than the  $\mathcal{O}(8^n)$  cost of  $U\rho U^\dagger$ . For  $r$  sequential applications, this yields  $\mathcal{O}(r \cdot 4^n)$  versus  $\mathcal{O}(r \cdot 8^n)$ —a factor of  $2^n$  improvement that dominates all other optimizations for larger system sizes.

A complementary optimization stores precomputed unitary matrices for each  $k$ -value rather than re-multiplying Pauli exponentials on each sample. This trades memory for a factor of  $K$  reduction in computation, where  $K$  is the truncation order. The trade-off is particularly favorable because the sampling distribution over  $k$ -values is highly non-uniform: lower-order terms dominate the Taylor coefficients, so their cached matrices are reused far more frequently than memory-equivalent uniform sampling would suggest.

Additional optimizations include vectorized probability calculations that compute all  $k$ -fold coefficient products in a single array operation rather than iterative loops, batch processing of samples grouped by  $k$ -values for improved cache locality, and strategic cache clearing between different simulation runs to bound memory usage while preserving intra-run benefits. Similar optimizations were applied to product formula methods: by computing the unitary matrix for a single Trotter step and then raising it to the power  $r$  via matrix exponentiation, we avoid the  $\mathcal{O}(r)$  cost of sequential multiplication, enabling simulations with CNOT gate counts exceeding  $10^{12}$ .

Since all operations in these circuits reduce to either Pauli string application or Pauli exponentials  $e^{-i\theta P}$ , the CNOT gate count can be computed deterministically without constructing the full circuit. For a Pauli string  $P = \bigotimes_j \sigma_j$  of weight  $w$  (number of non-identity factors), implementing  $e^{-i\theta P}$  requires  $2(w - 1)$  CNOT gates under a standard decomposition:  $w - 1$  CNOTs to compute the parity onto a single qubit, one single-qubit rotation, and  $w - 1$  CNOTs to uncompute. This closed-form expression allows us to tally CNOT counts across all Trotter steps or LCU terms by summing over the Pauli weights, enabling resource estimation for circuits far beyond what explicit construction would permit.

### 3.5.1 Resource Trade-offs

The simulation methods surveyed span a spectrum of resource trade-offs suited to different hardware regimes. Ancilla-free methods (Trotter, qDRIFT, THRIFT) operate entirely on the system register but incur circuit depths scaling polynomially with precision:  $\mathcal{O}(1/\varepsilon)$  for first-order formulas,  $\mathcal{O}(1/\varepsilon^{1/2k})$  for order- $2k$ . Single-ancilla LCU adds one qubit but achieves polylogarithmic depth  $\mathcal{O}(\log(1/\varepsilon)/\log\log(1/\varepsilon))$ , fitting comfortably within the  $\sim 10^3$ – $10^4$  depth ceiling characteristic of early fault-tolerant devices. Standard multi-ancilla LCU and quantum signal processing achieve near-optimal query complexity but require infrastructure (QROM, controlled arithmetic) beyond current capabilities.

All measurement-based expectation-value estimators share a fundamental sample complexity of  $\mathcal{O}(\|O\|^2/\varepsilon^2)$  arising from Hoeffding’s inequality, which cannot be circumvented without coherent amplitude amplification. The key distinction is total gate count: Trotter methods scale as  $\mathcal{O}(1/\varepsilon^3)$  (depth times samples), while single-ancilla LCU scales as  $\mathcal{O}(\log(1/\varepsilon)/\varepsilon^2)$ . For devices with 50–200 logical qubits and high precision requirements ( $\varepsilon < 10^{-4}$ ), single-ancilla LCU offers an optimal balance of depth, ancilla cost, and precision scaling; for smaller devices or moderate precision targets, ancilla-free product formulas remain competitive due to their simpler circuit structure. The following benchmarks quantify these trade-offs on two representative systems: a sparse nearest-neighbor Ising model where commutator bounds favor product formulas, and a dense quantum chemistry Hamiltonian where norm-based methods excel.

### 3.5.2 One-Dimensional Ising Model

The transverse-field Ising model with near-neighbor interactions serves as our primary scalable testbed. Throughout our benchmarks, we measure the average magnetization  $M_z = \frac{1}{N} \langle \sum_{i=1}^N Z_i \rangle$ , which represents the mean magnetic moment across all spins in the system. This system is defined by the Hamiltonian

$$H_{\text{Ising}} = - \sum_{i=1}^{N-1} \sigma_i^z \sigma_{i+1}^z + h \sum_{j=1}^N \sigma_j^x, \quad (3.5.1)$$

where  $N$  is the number of qubits and  $h$  is the transverse-field strength. This Hamiltonian is widely used in benchmarking quantum algorithms for several reasons. First, it exhibits a quantum phase transition at  $h = 1$ , allowing us to probe interesting physical regimes. Second, its sparse structure (each term commutes with all but at most two others) permits tight analytical bounds on product-formula methods, making it ideal for validating theoretical predictions. Third, the number of terms  $L = 2N - 1$  scales linearly with system size, enabling scalability studies. Finally, its one-dimensional geometry with nearest-neighbor interactions reflects the connectivity constraints of many near-term quantum devices.

The sparsity of this Hamiltonian allows us to derive tighter commutator bounds for first-order Trotterization. Following Childs et al. [61], the commutator bound for first-order Trotter on this system can be tightened to

$$\tilde{\alpha}_{\text{comm}} = 2h(N - 1), \quad (3.5.2)$$

yielding a segmentation parameter

$$r = \mathcal{O}\left(\frac{Nht^2}{\varepsilon}\right) \quad (3.5.3)$$

for error tolerance  $\varepsilon$ . This bound exploits the fact that most pairs of terms in the Hamiltonian commute.

We benchmark first-order and second-order Trotterization, qDRIFT, and single-ancilla LCU with truncated Taylor series on the one-dimensional Ising model defined in Eq. (3.5.1). Our primary metric is CNOT gate count, which we evaluate by varying system size  $N$ , evolution time  $t$ , and target error  $\varepsilon$ .

Figure 3.3 presents a comprehensive comparison for a 25-qubit Ising chain across the  $(t, \varepsilon)$  parameter space. The left panel shows CNOT gate count as a heatmap, while the right panel identifies which method achieves the lowest gate count in each regime. At large error tolerances ( $\varepsilon \gtrsim 10^{-2}$ ), first-order Trotterization benefits from the tight commutator bound in Eq. (3.5.3) and achieves the shallowest circuits. As precision requirements increase ( $\varepsilon < 10^{-3}$ ), single-ancilla LCU begins to dominate due to its favorable  $\mathcal{O}(\log(1/\varepsilon))$  scaling. Notably, qDRIFT shows no advantage for this sparse Hamiltonian: because the one-norm  $\lambda$  and the number of terms  $L$  scale identically ( $\lambda = L = 2N - 1$ ), the method's  $\lambda$ -dependence provides no benefit over term-counting approaches.

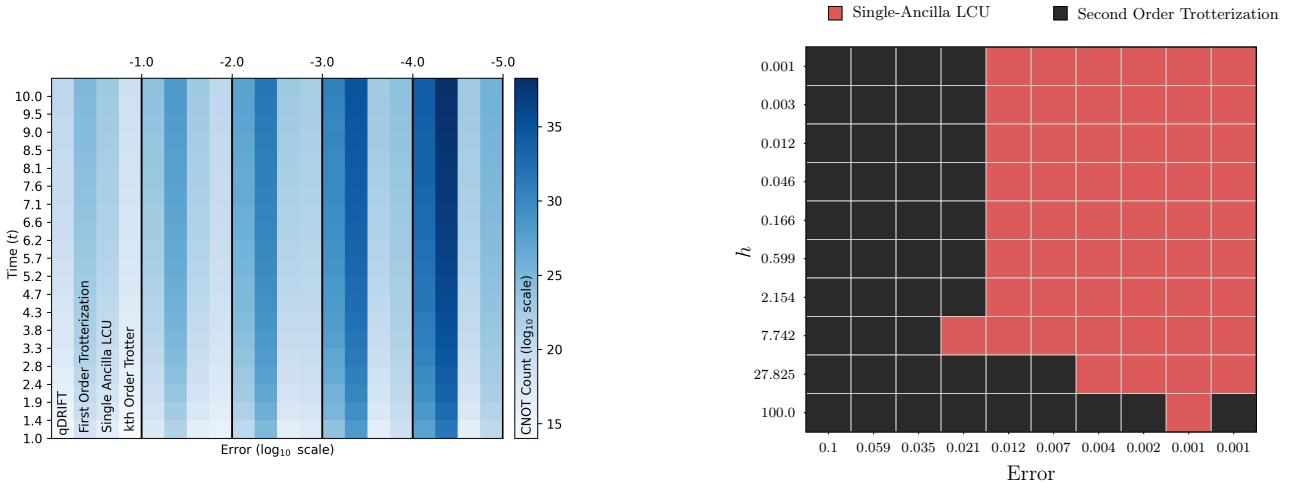


Figure 3.3: Benchmarking results for a 25-qubit one-dimensional Ising model. Left: CNOT gate count as a function of evolution time  $t$  and error  $\varepsilon$ . Right: Method with lowest gate count in each regime. Single-ancilla LCU dominates at high precision ( $\varepsilon < 10^{-3}$ ), while first-order Trotterization is preferable at larger error tolerances.

To isolate individual scaling dependencies, we examine how gate count varies with system size, evolution time, and target error. Figure 3.4 shows CNOT count versus  $N$  at fixed  $t = 1$  and  $\varepsilon = 10^{-3}$ : all methods exhibit

linear scaling, consistent with both  $L$  and  $\lambda$  growing linearly for this model. Single-ancilla LCU achieves the lowest gate counts across all system sizes due to its logarithmic error dependence.

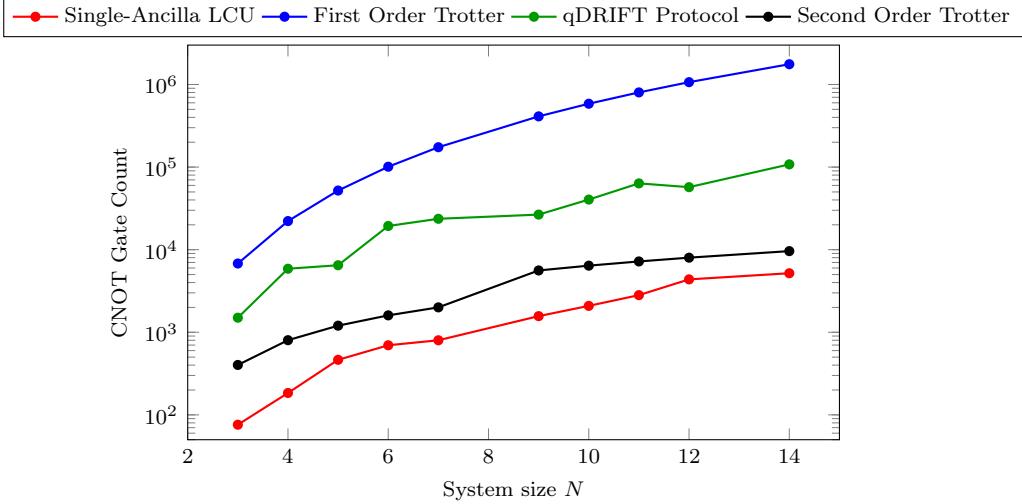


Figure 3.4: CNOT gate count versus system size  $N$  for the one-dimensional Ising model at fixed  $t = 1$  and  $\varepsilon = 10^{-3}$ . All methods scale linearly in  $N$ , with single-ancilla LCU achieving the lowest gate counts across all system sizes.

Figure 3.5 shows how gate count scales with evolution time (left) and target error (right) for a 25-qubit chain. For time scaling at fixed  $\varepsilon = 10^{-3}$ , second-order Trotterization outperforms single-ancilla LCU at longer times due to its favorable  $t^{3/2}$  scaling compared to SA-LCU’s  $t^2$  dependence. The error scaling plot (right panel, at fixed  $t = 10$ ) reveals the regime where SA-LCU excels: first-order Trotterization scales as  $\mathcal{O}(1/\varepsilon)$ , second-order as  $\mathcal{O}(1/\sqrt{\varepsilon})$ , and qDRIFT as  $\mathcal{O}(1/\varepsilon)$ , while single-ancilla LCU achieves  $\mathcal{O}(\log(1/\varepsilon))$ . This logarithmic scaling means that improving precision by an order of magnitude requires only a constant additive increase in gate count for LCU, compared to multiplicative increases for product formulas. At  $t = 10$ , crossover from second-order Trotter to SA-LCU occurs between  $\varepsilon = 10^{-3}$  and  $\varepsilon = 10^{-4}$ . At  $\varepsilon = 10^{-5}$ , single-ancilla LCU requires roughly  $9 \times 10^6$  CNOT gates compared to  $5 \times 10^7$  for second-order Trotter and  $3 \times 10^{11}$  for first-order Trotterization.

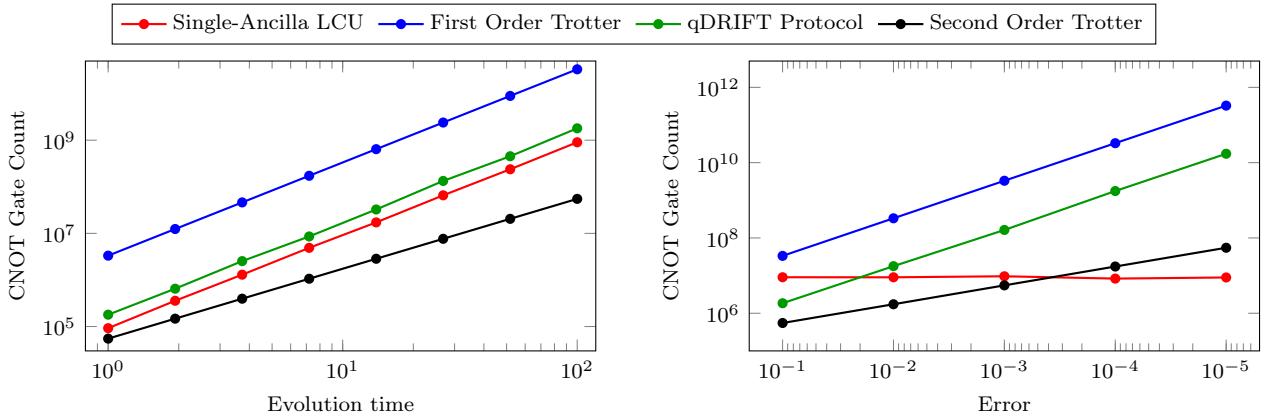


Figure 3.5: CNOT gate count for a 25-qubit Ising chain. Left: gate count versus evolution time  $t$  at fixed  $\varepsilon = 10^{-3}$ . Right: gate count versus target error  $\varepsilon$  at fixed  $t = 10$ . The logarithmic error scaling of single-ancilla LCU dramatically outperforms product formulas at high precision.

### 3.5.3 Quantum Chemistry

Inspired by the benchmarking methodology of Campbell [62], we also consider quantum chemistry Hamiltonians, which present significantly different characteristics than Ising models. We focus on the methane molecule ( $\text{CH}_4$ ), whose electronic-structure Hamiltonian is obtained from the Pennylane datasets [63]. These Hamiltonians are constructed by performing Hartree-Fock calculations to obtain molecular integrals, building the fermionic

Hamiltonian in second quantization, and mapping to qubit operators via the Jordan-Wigner transformation. Unlike the Ising model, where  $L = 2N - 1$  scales linearly (giving  $L = 49$  terms for  $N = 25$  qubits), quantum chemistry Hamiltonians exhibit quartic scaling with the number of spin-orbitals. For the 18-qubit methane Hamiltonian used here,  $L = 6,892$  terms, over 140 times more terms despite fewer qubits. This rapid growth means even modest molecules can have tens of thousands of terms (e.g.,  $\text{CO}_2$  with 30 qubits has  $L = 16,170$ ). Despite this large  $L$ , chemistry Hamiltonians typically have relatively small per-term coefficients, leading to a one-norm  $\lambda$  that grows more slowly than  $L$ . This property makes them ideal for demonstrating the advantages of methods like qDRIFT and single-ancilla LCU, which depend on  $\lambda$  rather than  $L$ . Additionally, chemistry Hamiltonians are dense (most terms do not commute), making tight analytical bounds for product formulas difficult to obtain and highlighting the value of norm-based simulation methods.

### Benchmarking Results

Figure 3.6 presents CNOT gate counts for methane as a function of evolution time  $t$  and error  $\varepsilon$ . The left panel shows a heat map of gate counts, while the right panel shows gate count versus time at fixed error. Single-ancilla LCU achieves significantly lower gate counts across nearly all regimes, with the advantage becoming more pronounced as  $\varepsilon$  decreases. qDRIFT also performs well due to its  $\lambda$ -dependence, consistently outperforming first-order Trotterization. Notably, even second-order Trotterization (not shown) struggles to compete in the displayed regime, as the dense structure of the chemistry Hamiltonian yields large commutator norms that weaken the effectiveness of higher-order product formulas. However, based on the time scaling slopes, second-order Trotter's favorable  $t^{3/2}$  dependence compared to SA-LCU's  $t^2$  scaling suggests it will likely prevail at sufficiently long evolution times.

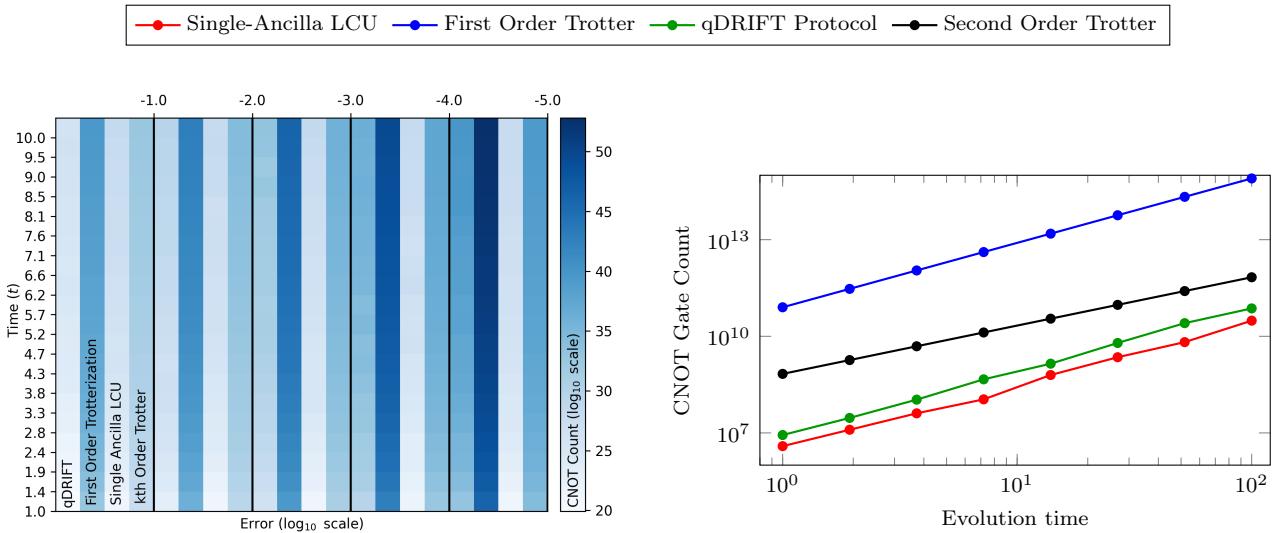


Figure 3.6: Benchmarking results for methane ( $\text{CH}_4$ ) Hamiltonian simulation. Left: CNOT gate count as a function of evolution time  $t$  and error  $\varepsilon$ . Right: Gate count versus time at fixed error  $\varepsilon = 10^{-3}$ . Single-ancilla LCU achieves the lowest gate counts across most regimes, demonstrating clear advantages for dense, large- $L$  Hamiltonians.

The error scaling for methane, shown in Figure 3.7, reveals an even more dramatic advantage for single-ancilla LCU than observed in the Ising model. At  $\varepsilon = 10^{-4}$ , single-ancilla LCU requires approximately  $1.5 \times 10^8$  CNOT gates, while first-order Trotterization exceeds  $10^{15}$ —a seven-order-of-magnitude difference. This stark contrast arises from the dense structure and large term count of the chemistry Hamiltonian. Notably, qDRIFT performs comparatively well on this system, achieving gate counts within two orders of magnitude of single-ancilla LCU, because its  $\lambda$ -dependence (rather than  $L$ -dependence) provides natural resilience to the large term count.

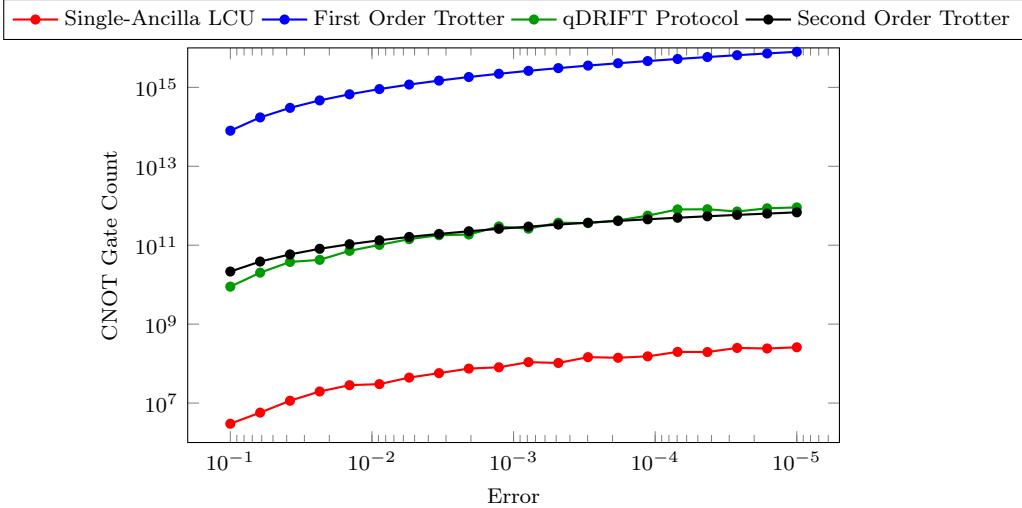


Figure 3.7: CNOT gate count versus target error  $\varepsilon$  for methane at fixed evolution time  $t = 10$ . The dense structure of the quantum chemistry Hamiltonian amplifies the advantage of single-ancilla LCU: its logarithmic error scaling (red) achieves a seven-order-of-magnitude improvement over first-order Trotterization at  $\varepsilon = 10^{-4}$ . qDRIFT (green) performs comparatively well due to its  $\lambda$ -dependence rather than  $L$ -dependence.

### 3.5.4 Summary

These benchmarks confirm that the optimal simulation method depends on system structure: for sparse Hamiltonians with tight commutator bounds, product formulas remain competitive at moderate precision, while single-ancilla LCU dominates at high precision ( $\varepsilon < 10^{-4}$ ) due to its logarithmic error scaling. For dense Hamiltonians where commutator bounds are ineffective, single-ancilla LCU and qDRIFT outperform product formulas by depending on the one-norm  $\lambda$  rather than the term count  $L$ . The classical simulation infrastructure developed in this section—memoization, sample deduplication, state-vector evolution, and matrix precomputation—will be reused in subsequent chapters to benchmark single-ancilla LCU applications to ground-state preparation (Chapter 4) and open quantum system simulation (Chapter 5).

## Chapter 4

# Applications: Ground State Preparation

This chapter presents ground state preparation using the Single-Ancilla Linear Combination of Unitaries (LCU) framework. We begin by establishing the motivation and problem formulation, then develop the theoretical foundations and implementation strategy, followed by complexity analysis and numerical benchmarks.

In quantum mechanics, the ground state  $|v_0\rangle$  of a Hamiltonian  $H$  is the eigenstate corresponding to the lowest eigenvalue  $E_0$ , satisfying  $H|v_0\rangle = E_0|v_0\rangle$  where  $E_0 \leq E_n$  for all eigenvalues [35]. The ground state represents the most stable configuration of a quantum system and may be unique (non-degenerate) or admit multiple linearly independent states sharing the minimum energy (degenerate). From statistical mechanics, systems at thermal equilibrium follow the Boltzmann distribution  $P(n) \propto e^{-E_n/k_B T}$ , and as temperature approaches zero, the probability of occupying the ground state approaches unity. Even at absolute zero, quantum fluctuations prevent the system from having exactly zero energy, a phenomenon known as zero-point energy.

The computational complexity of ground state problems is formalized through the  $k$ -local Hamiltonian problem: given a Hamiltonian  $H = \sum_i H_i$  where each term  $H_i$  acts non-trivially on at most  $k$  qubits, determine whether the ground state energy lies below a specified threshold. This problem is QMA-complete, the quantum analogue of NP-complete, meaning it is among the hardest problems verifiable by a quantum computer [48]. Kempe, Kitaev, and Regev [64] proved that even the 2-local Hamiltonian problem is QMA-complete, and Oliveira and Terhal [65] extended this to show that 2-local Hamiltonians on a 2D square lattice remain QMA-complete. The fundamental difficulty stems from the exponential growth of the Hilbert space: describing an  $n$ -qubit state classically requires  $2^n$  complex amplitudes, and highly entangled states, which are common in ground states of interacting systems, cannot be efficiently represented by classical means [66].

Several quantum algorithms have been proposed to address ground state preparation. The Variational Quantum Eigensolver (VQE) [6] is a hybrid quantum-classical algorithm that uses a parameterized quantum circuit (ansatz) to prepare trial states  $|\psi(\theta)\rangle$  and exploits the variational principle  $\langle\psi(\theta)|H|\psi(\theta)\rangle \geq E_0$  to iteratively minimize the energy. While VQE requires relatively shallow circuits suitable for noisy intermediate-scale quantum (NISQ) devices, it is fundamentally heuristic: the optimization landscape can exhibit barren plateaus where gradients vanish exponentially, and the algorithm may converge to local minima rather than the true ground state [67]. Quantum Phase Estimation (QPE) [39], by contrast, can determine ground state energies with provable guarantees by time-evolving a state with sufficient overlap with the ground state. This overlap assumption is generally acceptable since mean-field states such as Hartree-Fock often achieve high overlap with the correlated ground state for molecular systems [30], and even modest overlap incurs only polynomial overhead in the number of repetitions. However, QPE requires deep coherent circuits and multiple ancilla qubits, making it impractical for near-term quantum hardware. This leaves a gap: we need algorithms with *provable guarantees* that are also suitable for the *early fault-tolerant era*.

Ground state determination has practical importance in quantum chemistry, where electronic ground states govern molecular properties such as reaction rates and binding affinities [30, 45]. Current quantum devices have successfully simulated small molecules at chemical accuracy, though scaling to industrially relevant systems remains a significant challenge.

The Single-Ancilla LCU framework introduced in Chapter 3 provides a natural foundation for ground state property estimation. The key insight is to apply a Gaussian filter  $f(H) = e^{-tH^2}$  that exponentially suppresses excited state components (since  $e^{-t\lambda_i^2} \ll 1$  for  $\lambda_i \geq \Delta$ ) while preserving the ground state component (since  $e^{-t \cdot 0^2} = 1$  when the ground energy is shifted to zero) [68, 5]. This filtering operation is implemented using the same single-ancilla randomized sampling strategy developed for Hamiltonian simulation, inheriting its favorable properties: shallow per-run circuits, no multi-qubit controlled gates, and provable guarantees on accuracy.

We now formalize the ground state property estimation problem and establish the assumptions and con-

ventions that will be used throughout this chapter. Given a Hamiltonian  $H$  acting on  $n$  qubits, the ground state  $|v_0\rangle$  is the eigenstate corresponding to the lowest eigenvalue (ground energy)  $\lambda_0$ . We assume knowledge of a lower bound on the spectral gap  $\Delta$ , where  $|\lambda_1 - \lambda_0| \geq \Delta$  and  $\lambda_1$  is the first excited state energy. As defined in Chapter 1, the spectral gap quantifies how well-separated the ground state is from the rest of the spectrum and plays a crucial role in determining algorithm complexity. We assume access to an initial state  $|\psi_0\rangle$  that has non-negligible overlap with the ground state,  $|\langle\psi_0|v_0\rangle| = c_0 \geq \eta$ , which is a standard assumption in quantum algorithms for ground state preparation [5, 30]. In practice, such states can often be prepared using simple heuristics or mean-field approximations such as Hartree-Fock, and the parameter  $\eta$  captures how good our initial guess is, with better initial states leading to more efficient algorithms.

For algorithmic efficiency, we assume approximate knowledge of the ground energy to precision:

$$|E_0 - \lambda_0| \leq \varepsilon_g = \mathcal{O}\left(\frac{\Delta}{\sqrt{\log(1/\eta\varepsilon)}}\right) \quad (4.0.1)$$

This precision can be obtained using early fault-tolerant ground energy estimation algorithms (such as those by Lin-Tong [5] or others) without adding asymptotic overhead to our algorithm's complexity. By shifting the Hamiltonian to  $H - (E_0 - \varepsilon_g)I$ , we ensure the ground energy lies in  $[0, 2\varepsilon_g]$  while preserving the spectral gap. Without loss of generality, we normalize the spectrum to  $[0, 1]$ . If the actual spectrum is  $[0, E_{\max}]$ , we work with  $H/E_{\max}$ , which has spectral gap  $\Delta/E_{\max}$ , and algorithm complexities scale accordingly.

Our objective is to estimate expectation values of observables with respect to the ground state:

$$\langle v_0 | O | v_0 \rangle \quad (4.0.2)$$

for any measurable observable  $O$ , to additive accuracy  $\varepsilon$ . This is strictly more general than ground state preparation itself, as it allows us to extract useful information about the ground state without necessarily preparing it perfectly. However, a major drawback of this formulation is that it cannot be used as a subroutine if an algorithm only allows estimation of an observable rather than preparation of the ground state itself.

**Definition 4.0.1** (Ground State Property Estimation). *Given a Hamiltonian  $H$  with spectral gap  $\geq \Delta$  and ground energy known to precision  $\varepsilon_g$ , an initial state  $|\psi_0\rangle$  with  $|\langle\psi_0|v_0\rangle| \geq \eta$ , an observable  $O$  with  $\|O\| \leq 1$ , target accuracy  $\varepsilon$ , and failure probability  $\delta$ , output an estimate  $\hat{\mu}$  such that*

$$|\hat{\mu} - \langle v_0 | O | v_0 \rangle| \leq \varepsilon \quad (4.0.3)$$

with probability at least  $1 - \delta$ .

This formulation allows us to compute physical properties such as energy differences, correlation functions, and local observables without the need for full quantum state tomography of the ground state.

## 4.1 The Single-Ancilla LCU Algorithm for Ground State Property Estimation

The single-ancilla LCU algorithm achieves ground state property estimation by applying a Gaussian filter  $f(H) = e^{-tH^2}$  that exponentially suppresses excited state components while preserving the ground state. This filter is approximated as a linear combination of Hamiltonian simulation operators  $e^{-i\tau_j H}$  [9, 5], where each term corresponds to time evolution under the system Hamiltonian. Instead of coherently superposing all terms, the algorithm randomly samples pairs of evolution operators according to the LCU coefficients, applies them via a simple single-ancilla circuit, and uses classical post-processing to extract the desired expectation values.

Each term in the LCU decomposition requires implementing Hamiltonian simulation  $e^{-i\tau H}$ , which can be realized using either the truncated Taylor series approach from Chapter 3 or product formulas such as Trotterization. These strategies yield different resource tradeoffs, which we analyze in the complexity bounds and numerical benchmarks that follow. Rather than deriving the complete mathematical foundations of the filtering strategy, which involves Fourier analysis and Gaussian discretization techniques developed in prior work [9, 5], we present the algorithm directly and focus on its implementation and resource requirements.

**Theorem 4.1.1** (Ground State Property Estimation via Single-Ancilla LCU). *Let  $H$  be a Hamiltonian acting on  $n$  qubits with ground state  $|v_0\rangle$ , spectral gap  $\Delta > 0$ , and ground energy known to precision  $\varepsilon_g = \mathcal{O}(\Delta/\sqrt{\log(1/\eta\varepsilon)})$ . Let  $O$  be an observable with  $\|O\| \leq 1$ , and suppose we are given an initial state  $|\psi_0\rangle$  satisfying  $|\langle v_0|\psi_0\rangle| \geq \eta$  for some  $\eta \in (0, 1/\sqrt{2}]$ , which can be prepared with circuit depth  $\tau_{\psi_0}$ .*

Then for any  $\varepsilon, \delta \in (0, 1)$ , there exists a quantum algorithm using a single ancilla qubit that outputs an estimate  $\hat{\mu}$  satisfying

$$|\hat{\mu} - \langle v_0 | O | v_0 \rangle| \leq \varepsilon \quad (4.1.1)$$

with probability at least  $(1 - \delta)^2$ . The algorithm requires

$$T = \mathcal{O} \left( \frac{\|O\|^2 \ln(1/\delta)}{\varepsilon^2 \eta^4} \right) \quad (4.1.2)$$

classical repetitions, each with maximum evolution time

$$\tau_{\max} = \mathcal{O} \left( \Delta^{-1} \log \left( \frac{\|O\|}{\varepsilon \eta} \right) \right) \quad (4.1.3)$$

and circuit depth per repetition

$$D_{\text{run}} = 2\tau_{\max} + \tau_{\psi_0} + d_O, \quad (4.1.4)$$

where  $d_O$  is the depth required to measure the observable  $O$ . The total evolution time is

$$\tau_{\text{total}} = \tau_{\max} \cdot T = \tilde{\mathcal{O}} \left( \frac{\|O\|^2}{\Delta \varepsilon^2 \eta^4} \right). \quad (4.1.5)$$

Furthermore, the algorithm requires no multi-qubit controlled operations beyond those needed for Hamiltonian simulation, and provides exponential quantum speedup over classical methods for generic Hamiltonians.

The algorithm achieves these guarantees by applying the Gaussian filtering operator  $f(H) = e^{-tH^2}$ , which admits a linear combination of unitaries (LCU) decomposition. By Lemma 14 of [9], for any approximation error  $\gamma > 0$ :

$$\left\| e^{-tH^2} - \sum_{j=-M}^M c_j e^{-i\tau_j H} \right\| \leq \gamma \quad (4.1.6)$$

where  $c_j \in \mathbb{R}$  are Gaussian coefficients,  $\tau_j \in \mathbb{R}$  are evolution times, and the number of terms scales as  $M = \mathcal{O}(\sqrt{t \log(1/\gamma)})$ . For ground state preparation with target precision  $\varepsilon$ , initial overlap  $\eta$ , and observable  $O$ , we set  $\gamma = \varepsilon \eta^2 / (30 \|O\|)$ .

The key parameters of this decomposition are well-established. The filter time scales as  $t = \mathcal{O}(1/\Delta^2)$  with the spectral gap  $\Delta$ , and crucially, the LCU weight satisfies  $\|c\|_1 = \sum_j |c_j| = \mathcal{O}(1)$ , meaning the decomposition introduces only constant overhead independent of system size. The evolution times scale as  $\tau_j = \mathcal{O}(1/\Delta)$ , which determines the Hamiltonian simulation cost for each term. This bounded  $\ell_1$ -norm arises from discretizing a normalized Gaussian distribution via Fourier analysis [9, 5].

### Single-Ancilla Implementation via Randomized Sampling

Rather than preparing the filtered state explicitly, the algorithm estimates expectation values  $\langle v_0 | O | v_0 \rangle$  through repeated measurements on a simple quantum circuit using only one ancilla qubit. This reformulation exploits the observation that for many practical applications, explicit state preparation is unnecessary since we only need to extract statistical information about the ground state. Each run proceeds as follows:

**Algorithm 4** Single-Ancilla Ground State Property Estimation (One Run)

**Require:** Initial state  $|\psi_0\rangle$ , observable  $O$ , LCU coefficients  $\{c_j\}_{j=-M}^M$

**Ensure:** Measurement outcomes for estimator construction

1: **Sampling:** Draw two independent indices  $j_1, j_2 \in \{-M, \dots, M\}$  according to the distribution:

$$\Pr[j] = \frac{|c_j|}{\|c\|_1} \quad (4.1.7)$$

2: **State Preparation:** Prepare the system register in  $|\psi_0\rangle$  and the ancilla qubit in  $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

3: **Controlled Evolution:** Apply the controlled unitary:

$$X_{j_1}^{(c)} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes e^{-i\tau_{j_1} H} \quad (4.1.8)$$

4: **Anti-Controlled Evolution:** Apply the anti-controlled unitary:

$$Y_{j_2}^{(a)} = |0\rangle\langle 0| \otimes e^{-i\tau_{j_2} H} + |1\rangle\langle 1| \otimes I \quad (4.1.9)$$

5: **Joint Measurement:** Measure the observable  $X \otimes O$  on both ancilla and system registers, where  $X$  is the Pauli-X operator on the ancilla and  $O$  is the observable on the system, obtaining outcome  $\mu$

6: **return** Measurement outcome  $\mu$

The quantum circuit structure is illustrated in Figure 4.1, which follows the same pattern as the single-ancilla LCU framework introduced in Chapter 3 for Hamiltonian simulation. The ancilla is prepared in a superposition state, then controlled and anti-controlled evolution operators are applied back-to-back based on randomly sampled indices from the LCU decomposition. After both evolution operators have been applied, a joint measurement of the observable  $X \otimes O$  is performed on both the ancilla and system registers, where  $X$  denotes the Pauli-X operator acting on the ancilla qubit.

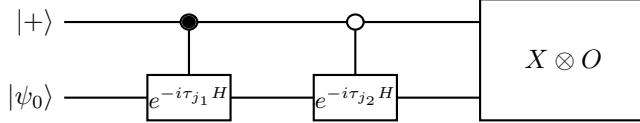


Figure 4.1: Single-ancilla LCU circuit for ground state property estimation. The ancilla (top) controls Hamiltonian evolution operators sampled from the LCU decomposition. The controlled and anti-controlled evolution operators are applied back-to-back, followed by a joint measurement of the observable  $X \otimes O$  on both the ancilla and system registers.

The key innovation is the use of randomized sampling: instead of coherently superposing all terms in the LCU decomposition, we randomly select indices  $j_1$  and  $j_2$  according to the importance sampling distribution  $|c_j|/\|c\|_1$ . This sampling scheme is similar to the single-ancilla approach used in Hamiltonian simulation (Chapter 3), where randomized sampling over LCU terms replaces coherent superposition to reduce ancilla requirements. The measurement outcomes across many runs produce unbiased estimators for the desired expectation values while maintaining bounded variance.

### Classical Post-Processing

The measurement outcomes collected across many independent runs of Algorithm 4 are used to construct two separate estimators through classical post-processing. These estimators approximate both the numerator and normalization of the quantity we wish to compute.

Let  $\mu_r$  denote the measurement outcome from the  $r$ -th run. By Theorem 8 of [9], the sample mean  $\hat{N}$  converges to  $\text{Tr}[O f(H)\rho_0 f(H)^\dagger]$  via Hoeffding's inequality. Running with  $O = I$  yields a normalization estimator  $\hat{D}$  for  $\ell^2 = \text{Tr}[f(H)\rho_0 f(H)^\dagger]$ .

The ground state expectation value is recovered via the ratio identity [5, 9]:

$$\langle v_0 | O | v_0 \rangle \approx \frac{\text{Tr}[O f(H)\rho_0 f(H)^\dagger]}{\text{Tr}[f(H)\rho_0 f(H)^\dagger]} \quad (4.1.10)$$

The final estimate for the ground state expectation value is obtained by taking the ratio:

$$\hat{\mu} = \frac{\hat{N}}{\hat{D}} \quad (4.1.11)$$

This ratio construction ensures convergence to  $\langle v_0 | O | v_0 \rangle$  even though neither estimator individually converges to the ground state expectation value. The normalization factor accounts for imperfect overlap between initial and filtered states.

Importance sampling according to  $|c_j|/\|c\|_1$  ensures bounded variance. By Theorem 9 of [9], the ratio  $\hat{\mu} = \hat{N}/\hat{D}$  satisfies  $|\hat{\mu} - \langle v_0 | O | v_0 \rangle| \leq \varepsilon$  with high probability. The complete correctness guarantee follows from Theorem 10 of [9].

## 4.2 Taylor Series Decomposition for Hamiltonian Evolution

Each term  $e^{-i\tau_j H}$  in the LCU decomposition of the Gaussian filter requires implementing Hamiltonian simulation. The truncated Taylor series approach from Chapter 3 provides a natural solution: the Taylor series expansion itself yields an LCU decomposition, enabling the same randomized sampling technique to be applied at this inner level. This nested structure preserves the single-ancilla constraint throughout the algorithm [9, 57].

For a Hamiltonian expressed as a weighted sum of Pauli operators  $H = \sum_\ell p_\ell P_\ell$ , the Taylor expansion produces unitaries that are products of Pauli strings followed by a single Pauli rotation. Each unitary in this decomposition takes the form  $W_m = (-i)^k P_{\ell_1} \cdots P_{\ell_k} e^{-i\theta P_m}$ , consisting of Clifford operations (the Pauli products) and one non-Clifford rotation. These elementary operations are ideally suited for single-ancilla control, as each can be implemented with standard controlled gates.

The coefficient norm of the Taylor series LCU can be made  $\mathcal{O}(1)$  through appropriate choice of the segmentation parameter [9]. This bounded norm is essential for preserving the efficiency of the single-ancilla framework, as it ensures that the sampling overhead from the inner LCU does not dominate the overall complexity.

The complete algorithm thus operates with a two-level sampling structure. At the outer level, each circuit run samples a term from the Gaussian filter LCU distribution according to  $|c_j|/\|c\|_1$ . At the inner level, it samples from the Taylor series LCU distribution to implement the required Hamiltonian evolution. Both levels use the same single ancilla qubit through randomized sampling rather than coherent superposition.

When each evolution operator  $U_j = e^{-i\tau_j H}$  is implemented approximately as  $\tilde{U}_j$ , the total error is bounded by  $\|c\|_1$  times the per-unitary precision. For target accuracy  $\varepsilon$  in measuring observable  $O$ , this yields a precision requirement of  $\mathcal{O}(\varepsilon/\|c\|_1 \|O\|)$  for each Hamiltonian simulation, which can be satisfied by the Taylor series parameters. The following theorem, based on Theorem A2 of [9], formalizes these guarantees.

**Theorem 4.2.1** (Imprecise Unitary Ground State Preparation). *Let  $\varepsilon, \delta \in (0, 1)$ ,  $O$  be some observable such that  $O = \sum_{j=1}^{L_O} h_j O_j$ , with  $\|h\|_1 = \sum_j |h_j|$ , and for any  $j$ ,  $\|O_j\| = 1$ . Also, let  $\rho_0$  be some initial state, prepared in cost  $\tau_{\rho_0}$ . Suppose  $H \in \mathbb{C}^{N \times N}$  be a Hermitian matrix such that for some function  $f : [-1, 1] \rightarrow \mathbb{R}$  and unitaries  $\{U_j\}_j$ ,*

$$\left\| f(H) - \sum_j c_j U_j \right\| \leq \frac{\varepsilon \ell_*}{27 \|h\|_1 \|f(H)\|},$$

and  $\ell^2 = \text{Tr}[f(H)\rho_0 f(H)^\dagger] \geq \ell_*^2$ . Moreover, suppose that each  $U_j$  can only be imperfectly implemented:  $\tilde{U}_j$  approximates  $U_j$  such that

$$\max_j \|U_j - \tilde{U}_j\| \leq \frac{\varepsilon \ell_*}{27 \|h\|_1 \|f(H)\| \|c\|_1}.$$

Furthermore, suppose that the maximum cost of implementing any  $\tilde{U}_j$  is at most  $\tau_{\max}$ .

Then there exists an algorithm that outputs  $\mu$  and  $\tilde{\ell}$  such that

$$\left| \frac{\mu}{\tilde{\ell}} - \text{Tr}[O\rho] \right| \leq \varepsilon,$$

with probability  $(1 - \delta)^2$ , using

$$T = \mathcal{O} \left( \frac{\|c\|_1^4 \|h\|_1^2}{\varepsilon^2 \ell_*^2} \ln(1/\delta) \right)$$

classical repetitions, where the cost of each such run is at most  $\mathcal{O}(2\tau_{\max} + \tau_{\rho_0})$ .

This is a reframing of Theorem A2 from [9].

The key advantage of expressing Hamiltonian evolution as an LCU via the truncated Taylor series is that it extends the single-ancilla framework without requiring Trotterization. The Taylor series decomposition provides a natural LCU structure that can be implemented with randomized sampling, maintaining the single-ancilla constraint throughout. This ancilla efficiency stems from two key properties of the algorithm, though it comes at the cost of preventing the algorithm from being used as a subroutine in more complex procedures.

First, the single-ancilla LCU algorithm implements each term in the decomposition through randomized sampling rather than coherent superposition. At each run, we randomly select one unitary  $W_m$  from the Taylor series decomposition according to the distribution  $\{|\alpha_m|/\|\alpha\|_1\}$ , then implement controlled and anti-controlled versions of this unitary using the single ancilla. The ancilla state is measured at the end of each run, resetting it for the next iteration. This measurement-based approach means the ancilla does not need to maintain coherent superposition across both levels of the decomposition.

Second, the imprecision introduced by truncating the Taylor series does not manifest as a requirement for additional quantum resources. Instead, it affects the number of terms  $M$  in the outer LCU decomposition needed to achieve the target accuracy. When we account for imprecise evolution operators, the effective outer LCU may require more terms to compensate for the accumulated error, but each term still uses only the single ancilla qubit.

The two-level structure can be summarized as follows: for each classical repetition of the algorithm, we:

1. Sample two independent terms  $j_1, j_2$  from the outer LCU distribution  $\{|c_j|/\|c\|_1\}$
2. For each selected term, sample inner unitaries  $W_{m_1}, W_{m_2}$  from the Taylor series distribution  $\{|\alpha_{jm}|/\|\alpha_j\|_1\}$
3. Implement the controlled version of  $W_{m_1}$  and the anti-controlled version of  $W_{m_2}$  using the single ancilla
4. Measure the ancilla and system to obtain an observable estimate

This process uses only one ancilla qubit throughout, regardless of the number of terms in either LCU decomposition.

The two-level structure introduces several modifications to the complexity analysis compared to assuming ideal Hamiltonian evolution oracles. The gate depth per coherent run becomes:

$$D_{\text{run}} = \mathcal{O}\left(\frac{\lambda^2}{\Delta^2}\right), \quad (4.2.1)$$

where  $\lambda = \sum_\ell |p_\ell|$  is the one-norm of Hamiltonian coefficients and  $\Delta$  is the spectral gap. This scaling reflects the cost of implementing the Taylor series decomposition for each Hamiltonian evolution operator in the outer LCU. The classical repetition overhead increases compared to using exact Hamiltonian simulation oracles, with the total number of circuit executions scaling as:

$$T = \mathcal{O}\left(\frac{\|h\|_1^2 \log(1/\delta)}{\varepsilon^2 \eta^4}\right), \quad (4.2.2)$$

where  $\|h\|_1$  is the one-norm of the observable coefficients,  $\eta$  is the initial state overlap with the ground state, and  $\delta$  is the failure probability. This overhead arises from the randomized sampling required to implement both the outer filtering LCU and the inner Hamiltonian evolution LCU. The key advantage lies in the ancilla requirement remaining fixed at one qubit regardless of the number of Hamiltonian terms  $L$  or the complexity of the Hamiltonian structure. For systems where  $\lambda \ll L$ , the gate depth scaling with  $\lambda^2$  rather than  $L$  provides significant practical advantages despite the increased sampling overhead.

This approach for ground state property estimation maintains minimal ancilla overhead throughout a hierarchical algorithmic structure. By using single-ancilla LCU for both the outer ground state filtering and the inner Hamiltonian simulation, the algorithm requires only one ancilla qubit in total. The imprecision introduced by approximate Hamiltonian evolution operators is absorbed through expanded LCU decompositions with more terms, rather than requiring additional quantum resources for error correction or amplitude amplification. This suits the constraints of early fault-tolerant quantum computers, where ancilla budget and controlled operation fidelity represent primary limiting factors. The increased classical repetition overhead represents a manageable trade-off in this regime, as classical processing and circuit preparation occur outside the quantum coherence requirements. While the quadratic scaling in  $\lambda$  and the  $\eta^{-4}$  repetition overhead may seem prohibitive in some regimes, the complete elimination of ancilla overhead beyond a single qubit and the independence from the number of Hamiltonian terms  $L$  can make this approach competitive for many practical applications, particularly in quantum chemistry where Hamiltonians typically have thousands of terms but relatively small coefficient norms.

The truncated Taylor series is not the only ancilla-free method suitable for implementing the imprecise unitaries required by Theorem 4.2.1. Product formulas such as Trotterization and randomized methods like qDRIFT, introduced in Chapter 3, provide alternative implementations with different complexity trade-offs. First-order Trotter has gate depth scaling as  $\mathcal{O}(L(\lambda/\Delta)^{1+1/2k})$  for  $2k$ -order formulas, while qDRIFT achieves  $\mathcal{O}(\lambda^2/\Delta^2)$  scaling independent of  $L$  through randomized compilation. The optimal choice depends on the specific Hamiltonian structure, target precision, and hardware constraints. Throughout this work, we assume a fully connected circuit topology. Section 4.4 presents comparative analysis and numerical simulations of these backends.

### 4.3 Comparative Resource Analysis

We compare the resource requirements of the single-ancilla LCU approach with alternative methods for ground state property estimation. We consider estimating  $\langle v_0 | O | v_0 \rangle$  to  $\varepsilon$ -additive accuracy, where  $|v_0\rangle$  is the ground state of a Hamiltonian  $H = \sum_{j=1}^L c_j P_j$  with one-norm  $\lambda = \sum_j |c_j|$  and spectral gap  $\Delta$ . The observable  $O = \sum_{k=1}^{L_O} h_k O_k$  has one-norm  $\|h\|_1 = \sum_k |h_k|$ . All algorithms assume access to an initial state with overlap at least  $\eta$  with  $|v_0\rangle$ .

Table 4.1: Resource comparison for ground state property estimation algorithms.

Algorithm	Gate depth per run	Classical repetitions
Standard LCU + QAA	$\tilde{\mathcal{O}}\left(\frac{\lambda L}{\Delta \eta}\right)$	$\mathcal{O}\left(\frac{\ h\ _1^2}{\varepsilon^2}\right)$
QSVT + QAE	$\tilde{\mathcal{O}}\left(\frac{\lambda L \ h\ _1}{\Delta \eta \varepsilon}\right)$	$\mathcal{O}(1)$
Single-Ancilla LCU	$\tilde{\mathcal{O}}\left(\frac{\lambda^2}{\Delta^2}\right)$	$\mathcal{O}\left(\frac{\ h\ _1^2}{\varepsilon^2 \eta^4}\right)$

Table 4.1 compares three approaches for ground state property estimation. Standard LCU with quantum amplitude amplification (QAA) [5] first prepares the ground state via filtering, then samples the observable, requiring  $\mathcal{O}(\log L)$  ancilla qubits. Quantum singular value transformation (QSVT) with quantum amplitude estimation (QAE) [58] achieves constant repetitions through coherent estimation but requires deeper circuits. The single-ancilla LCU approach [9] uses only one ancilla qubit throughout.

Without amplitude amplification techniques, Standard LCU and QSVT have identical baseline complexity: gate depth  $\tilde{\mathcal{O}}(\lambda L/\Delta)$  with  $\mathcal{O}(\|h\|_1^2/\varepsilon^2 \eta^2)$  repetitions. QAA moves the  $\eta$  dependence from repetitions into gate depth, while QAE eliminates repetition overhead entirely at the cost of  $\varepsilon$ -dependent circuit depth.

The single-ancilla approach trades quadratic gate depth scaling  $\mathcal{O}(\lambda^2/\Delta^2)$  versus linear  $\mathcal{O}(\lambda L/\Delta)$  for complete independence from the number of Hamiltonian terms  $L$ . This makes it advantageous for sparse Hamiltonians where  $\lambda \ll L$ , which is common in quantum chemistry applications with thousands of terms but moderate coefficient norms. The  $\eta^{-4}$  repetition overhead compared to  $\eta^{-2}$  for standard methods remains manageable when reasonable initial state overlap is available, which is achievable via mean-field methods such as Hartree-Fock that add minimal circuit overhead. The reduction from  $\mathcal{O}(\log L)$  to a single ancilla qubit represents the primary practical advantage for early fault-tolerant devices.

### 4.4 Numerical Simulations

This section extends the benchmarking framework developed in section 3.5 to the ground state property estimation problem. Following the reduction of ground state preparation to sampling from a linear combination of Hamiltonian simulations via the single-ancilla LCU method, we apply similar benchmarking techniques to compare circuit performance across different Hamiltonian simulation backends. The key difference in this context is that the Hamiltonian simulation unitaries are implemented as controlled operations through the single ancilla. Since the entire controlled Hamiltonian simulation must execute coherently without intermediate measurements, shallower circuits with higher controlled-gate counts typically outperform longer circuits, as reduced depth minimizes decoherence during the coherent execution window.

Beyond the optimizations inherited from Hamiltonian simulation, ground state property estimation introduces additional computational efficiencies specific to the single-ancilla LCU framework. The method requires implementing controlled and anti-controlled unitaries sequentially on the ancilla qubit. Our implementation exploits this structure through a two-stage post-selection strategy: the first controlled evolution (control value

1) is applied and its resulting state is cached, while the second stage chains this with the anti-controlled evolution (control value 0). Since many sample pairs share the same first index, memoizing the intermediate state after the first stage avoids redundant computation, with caching ensuring each unique first-stage evolution is computed exactly once. A complementary optimization exploits the repetition structure inherent in LCU sampling: when drawing  $N$  sample pairs  $(j_1, j_2)$  from the coefficient distribution, many pairs are identical due to non-uniform probability weights. Using sample clustering, we group identical pairs and compute each unique post-selection outcome exactly once, weighting results by multiplicity. This typically reduces expensive state evolution calculations by 70–80%: for instance, 1000 independent samples collapse to approximately 200 unique pairs.

#### 4.4.1 Ising Model

We benchmark ground state preparation for the one-dimensional transverse-field Ising model defined in [Equation 3.5.1](#). Unlike Hamiltonian simulation where evolution time is a variable parameter, ground state preparation does not have adjustable evolution time for a fixed target precision. Instead, we measure the scalability of different methods based on decreasing error rates.

[Figure 4.2](#) demonstrates the slow growth in CNOT gate count of the single-ancilla LCU method for a 25-qubit Ising model as the target error decreases. The controlled operations required by the single-ancilla framework are implemented using various Hamiltonian simulation techniques (first-order Trotterization, qDRIFT, and truncated Taylor series), and we observe that the optimal choice of simulation backend depends on the target precision regime.

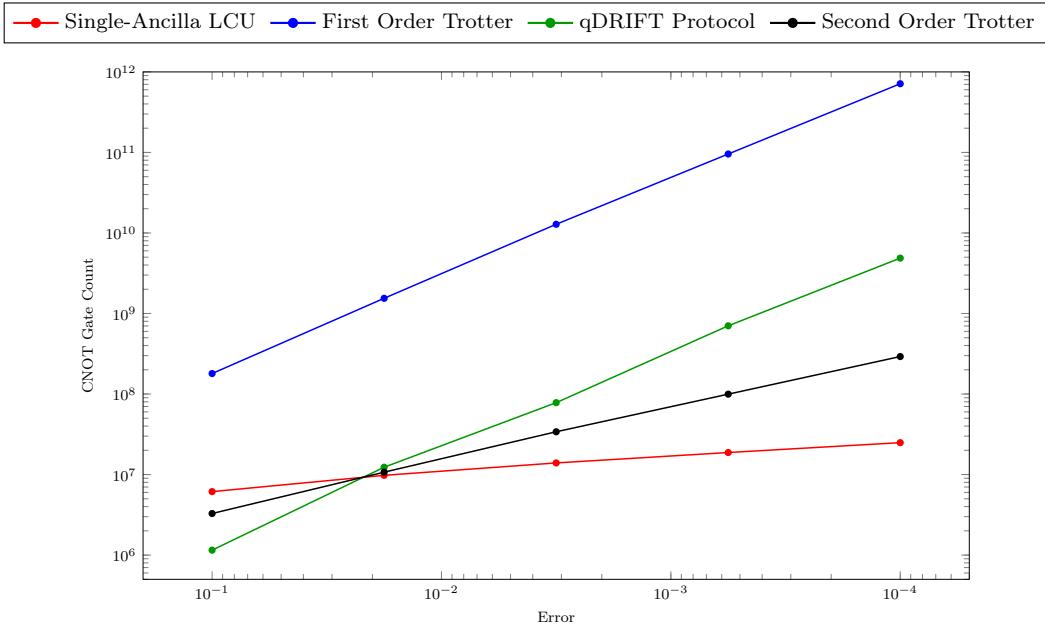


Figure 4.2: CNOT gate count scaling for ground state preparation of the 25-qubit transverse-field Ising model using single-ancilla LCU with different Hamiltonian simulation backends. The single-ancilla LCU method maintains relatively low gate counts across the error range.

At higher error tolerances, simple product formulas remain competitive due to their minimal overhead per Trotter step. However, as precision requirements tighten, methods with better asymptotic error scaling become advantageous despite their increased per-step cost. The crossover point depends on the specific Hamiltonian structure and commutator bounds.

To validate the single-ancilla LCU method for ground state property estimation, we measure the average magnetization  $M_z = \frac{1}{N} \langle \sum_{i=1}^N Z_i \rangle$  across the quantum phase transition of the transverse-field Ising model. [Figure 4.3](#) shows the magnetization as a function of the transverse field strength  $h$  for 8, 10, and 12-qubit chains using exact diagonalization, following the analytical methodology of [69].

The results demonstrate the characteristic quantum phase transition of the transverse-field Ising model. At low field ( $h \ll 1$ ), the ground state exhibits ferromagnetic order with  $M_z \approx 1$ , while at high field ( $h \gg 1$ ), the paramagnetic phase dominates with reduced magnetization. The transition region near  $h \approx 1$  shows finite-size effects: larger systems ( $N=12$ ) exhibit sharper transitions compared to smaller systems ( $N=8$ ), reflecting

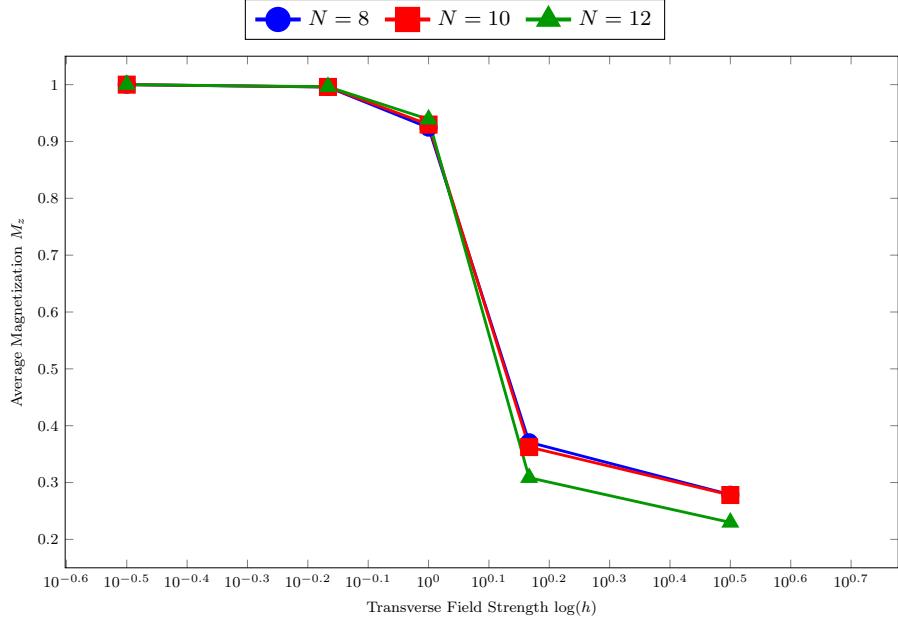


Figure 4.3: Ground state magnetization  $M_z$  versus transverse field strength  $h$  for 8, 10, and 12-qubit Ising chains obtained via exact diagonalization. The quantum phase transition occurs near  $h \approx 1$ , with larger systems exhibiting sharper transitions toward the thermodynamic limit.

convergence toward the thermodynamic limit where the transition becomes discontinuous.

#### 4.4.2 Quantum Chemistry

Ground state property estimation for molecular systems is an application of quantum computing with implications for drug discovery, materials design, and catalysis. We benchmark the single-ancilla LCU method on the methane molecule ( $\text{CH}_4$ ), a prototypical quantum chemistry system where the electronic Hamiltonian exhibits dense coupling between orbitals. For the observable, we use the Hamiltonian itself ( $O = H$ ), as ground state energy estimation is the primary objective in quantum chemistry applications and forms the basis of variational algorithms such as VQE. Since the measurement overhead scales as  $\|O\|^2/\varepsilon^2$  uniformly across all simulation methods, this choice does not affect the relative performance comparison.

Figure 4.4 shows the CNOT gate count scaling for ground state preparation of the methane molecule as the target error decreases. Quantum chemistry Hamiltonians typically have many terms, making the  $L$ -independence of the single-ancilla LCU approach particularly advantageous compared to product formula methods. The controlled Hamiltonian simulation backends demonstrate varying performance across different precision regimes, with qDRIFT and Taylor series methods showing favorable scaling at higher precision requirements.

These results demonstrate the single-ancilla LCU approach for quantum chemistry applications, where the combinatorial growth in Hamiltonian terms would otherwise impose prohibitive overhead on Trotterization-based methods. The ability to achieve ground state property estimation with gate counts scaling favorably in both system size and precision may enable near-term quantum advantage in molecular simulation. The inner-level Hamiltonian simulation can be implemented using either the single-ancilla LCU approach with Taylor series decomposition or product formula methods such as Trotterization. Single-ancilla LCU scales independently of the number of Hamiltonian terms  $L$ , making it advantageous for dense Hamiltonians like those in quantum chemistry. Trotterization, while exhibiting explicit  $L$ -dependence, requires no additional ancilla qubits and can outperform LCU for sparse Hamiltonians with favorable commutator structure.

Notably, even when using tight commutator bounds (alpha commutator bounds that exploit the specific commutation structure of the Hamiltonian rather than worst-case estimates), single-ancilla LCU maintains its advantage for ground state preparation. This is because the controlled operations required by the single-ancilla framework impose additional overhead on product formula methods, and the  $L$ -independence of LCU becomes increasingly valuable as system complexity grows. The optimal choice ultimately depends on the specific Hamiltonian structure, target precision, and hardware constraints. The numerical benchmarks in the following section quantify these trade-offs for representative systems.

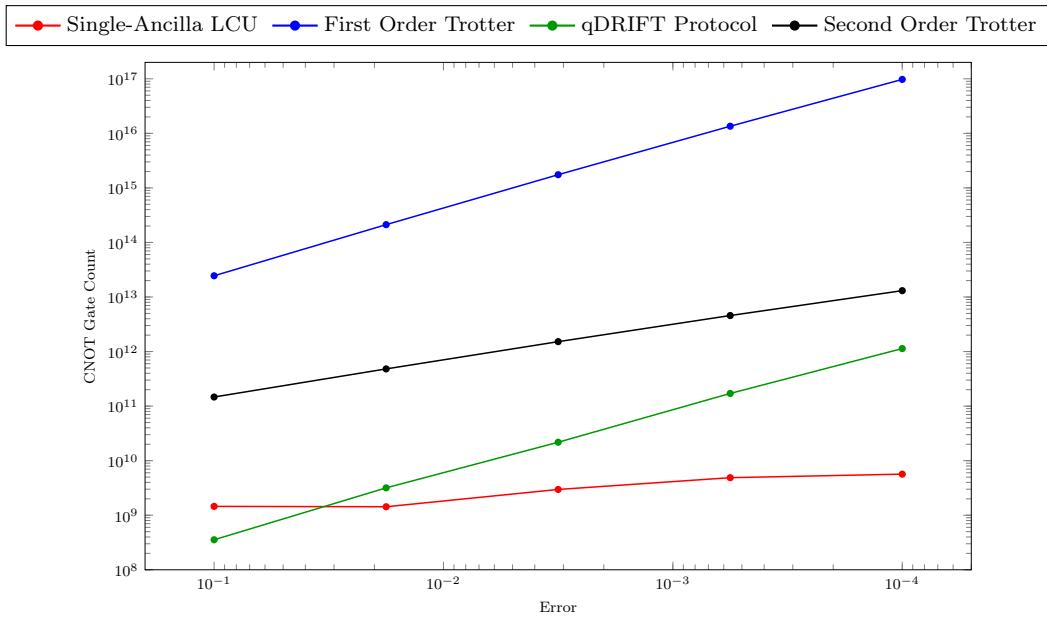


Figure 4.4: CNOT gate count scaling for ground state preparation of the methane molecule using single-ancilla LCU with different Hamiltonian simulation backends. The dense coupling structure of quantum chemistry Hamiltonians favors methods with reduced dependence on the number of Hamiltonian terms.

# Chapter 5

## Applications: Simulating quantum collision models

### 5.1 Introduction

This chapter presents the simulation of open quantum systems using quantum collision models, also known as repeated interaction schemes. We develop quantum algorithms that leverage the Hamiltonian simulation techniques from [chapter 3](#) to implement both Markovian and non-Markovian dynamics on early fault-tolerant quantum computers.

While simulating the dynamics of closed quantum systems (Hamiltonian simulation) is widely considered one of the foremost applications of quantum computers [20], real quantum systems inevitably interact with their environment, resulting in non-unitary open system dynamics [70]. The most widely studied framework is the Gorini-Kossakowski-Sudarshan-Lindblad (GKSL) master equation, which models Markovian quantum dynamics where the system’s future evolution depends only on its present state [71, 72]. This framework generates completely positive trace-preserving (CPTP) maps that capture essential phenomena including spontaneous emission, dephasing, and dissipation. Lindbladian dynamics has diverse applications across quantum technologies, including quantum error mitigation, state preparation, quantum thermodynamics, and quantum optics.

While quantum computers should provide advantages for simulating open systems due to favorable scaling with Hilbert space dimension, current state-of-the-art methods face significant barriers for early fault-tolerant implementation. Most methods require access to specialized oracles such as block encodings, multiple ancilla qubits, and sophisticated controlled operations [73]—resources ill-suited for the early fault-tolerant era [18]. Quantum collision models address these challenges [33]. In this framework, the environment comprises several individual subsystems (sub-environments), each interacting with the system sequentially for fixed time intervals. The interaction between system and sub-environment is simply time evolution under the total Hamiltonian, allowing any Hamiltonian simulation algorithm to be incorporated. This provides a unified framework that can model both Markovian dynamics (memoryless collisions) and non-Markovian dynamics (memory-retaining collisions with inter-environment interactions).

In the following sections, we establish the Markovian collision model formalism, where independent environmental subsystems interact sequentially with the system. Special attention is given to implementing these dynamics using the Single-Ancilla LCU algorithm, where we show how to avoid exponential scaling issues that arise when composing linear combinations of unitaries. We then extend the framework to non-Markovian dynamics, where environmental subsystems can interact with each other, enabling memory effects. A key application is the simulation of Lindbladian master equations, where we demonstrate how collision models provide a natural implementation pathway. Finally, we provide comprehensive complexity analyses and numerical benchmarks that compare Trotterization, qDRIFT, and SA-LCU approaches across various physically relevant scenarios, offering practical guidance for early fault-tolerant implementations.

### 5.2 Simulating Markovian Collision Models

Our goal is to estimate observables for a quantum system that undergoes repeated interactions with independent environmental subsystems. Specifically, we aim to develop quantum algorithms that can estimate the expectation value  $\text{Tr}[O \rho_{\text{final}}]$  of an observable  $O$  after the system has undergone  $K$  discrete collisions with the

environment, achieving additive precision  $\varepsilon$ .

The system to be simulated consists of an  $n$ -qubit quantum system  $\mathcal{H}_S$  that interacts sequentially with  $K$  independent environmental subsystems. Each sub-environment  $E_j$  lives in Hilbert space  $\mathcal{H}_{E_j}$ . During the  $j$ -th collision, the system and sub-environment evolve under the total Hamiltonian:

$$H_j = H_S + H_{E_j} + H_{I_j} = \sum_{i=1}^{L_j} h_{i,j} P_{i,j}, \quad (5.2.1)$$

where  $H_S$  is the system Hamiltonian,  $H_{E_j}$  is the sub-environment Hamiltonian,  $H_{I_j}$  is the interaction Hamiltonian,  $P_{i,j}$  are Pauli operators, and  $h_{i,j} \in \mathbb{R}^+$  are coefficients. Let  $L = \max_j L_j$  denote the maximum number of terms.

We define  $\bar{H}_j = H_j/\lambda_j$  where  $\lambda_j = \sum_{i=1}^{L_j} h_{i,j}$ , allowing us to rescale collision time  $\Delta t$  accordingly. Each collision evolves under:

$$U_j = e^{-i\lambda_j \bar{H}_j \Delta t}. \quad (5.2.2)$$

After each collision lasting time  $\Delta t$ , the sub-environment is discarded (traced out), leaving the system in a transformed state. Since each sub-environment interacts only once with no memory of previous collisions, this defines a Markovian collision map:

**Definition 5.2.1** (Markovian Collision Map). *The  $j$ -th collision map  $\Phi_j$  is defined as:*

$$\Phi_j[\cdot] \equiv \text{Tr}_{E_j} [U_j (\cdot \otimes \rho_{E_j}) U_j^\dagger], \quad (5.2.3)$$

where  $\text{Tr}_{E_j}$  denotes the partial trace over the  $j$ -th sub-environment.

After  $K$  collisions, the cumulative effect is described by the  $K$ -collision map:

**Definition 5.2.2** (Markovian  $K$ -Collision Map). *The  $K$ -collision map  $\mathcal{M}_K$  is the composition of  $K$  sequential collision maps:*

$$\mathcal{M}_K[\cdot] \equiv [\bigcirc_{j=1}^K \Phi_j[\cdot]] = \Phi_K \circ \Phi_{K-1} \circ \cdots \circ \Phi_1[\cdot]. \quad (5.2.4)$$

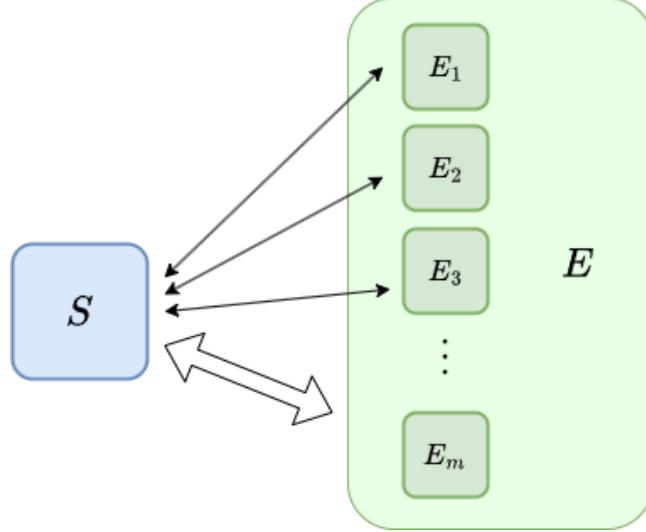


Figure 5.1: Schematic representation of the collision model depicting interactions between a quantum system  $S$  (blue) and an ordered sequence of environmental subsystems  $E_0, E_1, \dots, E_{m-1}$  (green). The system undergoes  $K$  sequential collisions, where each collision involves interaction with one environmental subsystem followed by tracing out that subsystem, implementing the Markovian  $K$ -collision map  $\mathcal{M}_K$ .

We approximate  $\mathcal{M}_K$  by implementing each collision unitary  $U_j$  using Hamiltonian simulation algorithms. These produce approximations  $\tilde{U}_j$  satisfying  $\|U_j - \tilde{U}_j\| \leq \varepsilon'$ , which yield approximate collision maps  $\tilde{\Phi}_j[\cdot] = \text{Tr}_{E_j} [\tilde{U}_j (\cdot \otimes \rho_{E_j}) \tilde{U}_j^\dagger]$ . Composing these gives the approximate  $K$ -collision map:

$$\widetilde{\mathcal{M}}_K[\cdot] = \bigcirc_{j=1}^K \tilde{\Phi}_j[\cdot]. \quad (5.2.5)$$

To estimate  $\text{Tr}[O \mathcal{M}_K[\rho_S]]$  to precision  $\varepsilon$ , we need each Hamiltonian simulation to satisfy:

**Lemma 5.2.3** (Bounds on the Markovian Collision Map). *Let  $O$  be an observable and  $\widetilde{\mathcal{M}}_K$  represent the approximate Markovian  $K$ -collision map in (5.2.5), where*

$$\max_{1 \leq j \leq K} \|U_j - \widetilde{U}_j\| \leq \frac{\varepsilon}{3K\|O\|}. \quad (5.2.6)$$

*Then, the expectation value of  $O$  over a state evolved under the approximate map satisfies:*

$$|\mathrm{Tr}[O \mathcal{M}_K[\rho]] - \mathrm{Tr}[O \widetilde{\mathcal{M}}_K[\rho]]| \leq \varepsilon. \quad (5.2.7)$$

*Proof.* The proof can be found in Ref. [8].  $\square$

If a Hamiltonian simulation algorithm requires depth  $\tau(\Delta t, \varepsilon')$  to implement each collision to precision  $\varepsilon' = \varepsilon/(3K\|O\|)$ , the total circuit depth is:

$$\tau_d = \mathcal{O}\left(K \cdot \tau\left(\Delta t, \frac{\varepsilon}{K\|O\|}\right) + K\tau_{\rho_E}\right), \quad (5.2.8)$$

where  $\tau_{\rho_E}$  is the maximum depth for preparing sub-environment states. Any Hamiltonian simulation method (Trotter, qDRIFT, SA-LCU, etc.) can be used.

### 5.3 Single-Ancilla LCU Implementation for Collision Models

While any near-term Hamiltonian simulation procedure can be used to implement the Markovian  $K$ -collision map, the Single-Ancilla LCU (SA-LCU) algorithm requires careful consideration due to potential exponential scaling when composing linear combinations of unitaries. In this section, we show how to avoid this exponential overhead and implement the  $K$ -collision map efficiently using SA-LCU.

To understand the challenge, we first recall the LCU framework from chapter 3. For a Hamiltonian  $H = \sum_{\ell=1}^L p_\ell P_\ell$  expressed as a convex combination of Pauli operators  $P_\ell$ , the time evolution operator  $U = e^{-i\tau H}$  can be approximated by a linear combination of unitaries (LCU). Specifically, we construct  $\widetilde{U} = \sum_i \alpha_i W_i$  that approximates  $U$  to precision  $\varepsilon$ , where each  $W_i$  is a sequence of  $g$  Clifford operators followed by a single Pauli rotation, repeated  $r$  times. The total weight  $\alpha = \sum_i |\alpha_i| \leq e^{\tau^2/r}$ , the parameter  $r$  is chosen such that  $\tau/r < 1$  (ensuring series convergence), and the truncation parameter  $q = \mathcal{O}(\log(r/\varepsilon)/\log\log(r/\varepsilon))$  controls the precision. For  $r = \mathcal{O}(\tau^2)$ , we have  $\alpha = \mathcal{O}(1)$ , and the LCU can be implemented using a single ancilla qubit with circuit depth  $\mathcal{O}(rq) = \widetilde{\mathcal{O}}(\tau^2)$  and  $T = \mathcal{O}(\|O\|^2/\varepsilon^2)$  classical repetitions.

However, when implementing a  $K$ -collision map by composing  $K$  SA-LCU Hamiltonian simulations, a naive approach would lead to an exponential scaling problem. If each collision  $j$  is implemented with  $\widetilde{U}_j = \sum_i \alpha_{ji} W_{ji}$  having total weight  $\alpha^{(j)} = \mathcal{O}(1)$ , then composing  $K$  such LCUs would yield a total weight of  $(\alpha^{(j)})^K$ , which grows exponentially with  $K$ . This exponential scaling would detrimentally affect both the circuit depth and the number of classical repetitions required for observable estimation.

To resolve this problem, we choose a different value of the parameter  $r$  that balances the LCU weight across the entire composition. Specifically, we modify the choice of parameter  $r_j$  for each collision. Recall from Lemma 5.2.3 that we need each Hamiltonian simulation to satisfy:

$$\|U_j - \widetilde{U}_j\| \leq \frac{\varepsilon}{3K\|O\|}, \quad (5.3.1)$$

where  $U_j = e^{-i\lambda_j \Delta t \overline{H}_j}$ .

For the LCU decomposition of each  $\widetilde{U}_j$ :

$$\widetilde{U}_j = \sum_k \alpha_{jk} W_{jk}, \quad (5.3.2)$$

we choose  $r_j$  such that:

$$\alpha^{(j)} = \sum_k |\alpha_{jk}| \leq e^{\lambda_j^2 \Delta t^2 / r_j^2}. \quad (5.3.3)$$

The key is to choose  $r_j = \mathcal{O}(\lambda_j \Delta t K)$  rather than  $r_j = \mathcal{O}(\lambda_j^2 \Delta t^2)$ . This ensures:

$$\alpha^{(j)} = \mathcal{O}\left(e^{\lambda_j^2 \Delta t^2 / (\lambda_j \Delta t K)^2}\right) = \mathcal{O}\left(e^{1/K^2}\right) = \mathcal{O}(1), \quad (5.3.4)$$

and the product over all collisions satisfies:

$$\zeta = \prod_{j=1}^K \alpha^{(j)} = \mathcal{O}(1). \quad (5.3.5)$$

With this choice, each  $W_{jk}$  consists of  $q$  Clifford operators and a single Pauli rotation, repeated  $r_j$  times, where:

$$q = \mathcal{O}\left(\frac{\log(r_j K \|O\|/\varepsilon)}{\log \log(r_j K \|O\|/\varepsilon)}\right). \quad (5.3.6)$$

With these parameter choices established, we can now present the complete algorithm for estimating observables after a  $K$ -collision map using SA-LCU.

---

**Algorithm 5** Estimate Observable After  $K$ -Collision Map Using SA-LCU

**Require:** Initial system state  $\rho_S$ , sub-environment states  $\{\rho_{E_j}\}_{j=1}^K$ , observable  $O$ , unitaries  $\{\tilde{U}_j\}_{j=1}^K$  with LCU decompositions  $\tilde{U}_j = \sum_k \alpha_{jk} W_{jk}$  satisfying  $\|\tilde{U}_j - e^{-i\Delta t \lambda_j H_j}\| \leq \varepsilon/(3K\|O\|)$ , precision  $\varepsilon$ , confidence  $\delta$

- 1: Initialize system register in  $\rho_S$  and ancilla in  $|+\rangle$
- 2: **for**  $j = 1$  to  $K$  **do**
- 3:     Initialize environment register in  $\rho_{E_j}$
- 4:     Draw i.i.d. samples  $X_j, Y_j$  from ensemble  $\mathcal{D}_j = \{W_{jk}, \alpha_{jk}/\alpha^{(j)}\}$
- 5:     Apply controlled  $X_j^{(c)}$  and anti-controlled  $Y_j^{(a)}$  to system-environment
- 6:     Trace out environment register
- 7: **end for**
- 8: Measure ancilla-system on observable  $\sigma^x \otimes O$ , record outcome  $\mu_i$
- 9: Repeat the entire process  $T = \mathcal{O}\left(\|O\|^2 \log(1/\delta)/\varepsilon^2\right)$  times
- 10: Compute estimate:  $\mu = \frac{\zeta^2}{T} \sum_{i=1}^T \mu_i$ , where  $\zeta = \prod_{j=1}^K \alpha^{(j)}$

**Ensure:** Estimate  $\mu$  satisfying  $|\mu - \text{Tr}[O \mathcal{M}_K[\rho_S]]| \leq \varepsilon$  with probability  $\geq 1 - \delta$

---

The resource requirements for Algorithm 5 are as follows. To estimate  $\text{Tr}[O \mathcal{M}_K[\rho_S]]$  to precision  $\varepsilon$  with probability  $\geq 1 - \delta$ , the algorithm requires:

$$T = \mathcal{O}\left(\frac{\|O\|^2 \log(1/\delta)}{\varepsilon^2}\right) \quad (5.3.7)$$

repetitions, where each coherent run has circuit depth:

$$\tau_d = \mathcal{O}\left(\lambda^2 K^2 \Delta t^2 \frac{\log(\lambda K \|O\| \Delta t / \varepsilon)}{\log \log(\lambda K \|O\| \Delta t / \varepsilon)} + K \tau_{\rho_E}\right), \quad (5.3.8)$$

where  $\lambda = \max_j \lambda_j$  and  $\tau_{\rho_E} = \max_j \tau_{\rho_{E_j}}$  is the maximum environment state preparation depth. The algorithm requires  $n + 2$  qubits total (system + environment + ancilla).

This approach successfully avoids exponential scaling in the LCU composition, making SA-LCU a viable method for implementing multi-collision dynamics on early fault-tolerant quantum computers.

## 5.4 Extension to Non-Markovian Dynamics

The Markovian collision model framework developed in previous sections assumes memoryless collisions: each sub-environment interacts with the system once before being discarded, with no information transfer between sub-environments. However, realistic open quantum systems often exhibit memory effects, where past interactions influence future dynamics. In this section, we extend the collision model framework to simulate non-Markovian dynamics by incorporating interactions between successive sub-environments.

### 5.4.1 Memory-Retaining Collision Map

Non-Markovian dynamics arise naturally when collisions are memory-retaining. Within the collision model framework, this manifests through inter-environment interactions that allow information from previous collisions to persist and influence subsequent system-environment interactions [33].

Two main mechanisms produce non-Markovianity in collision models: system-environment feedback, where information flows back from the environment to the system across multiple collisions, and inter-environmental correlations, where sub-environments interact with each other to create memory channels.

We focus on the second mechanism, following the framework of Ciccarello et al. [74], where collisions between consecutive sub-environments enable memory effects. These inter-environmental correlations establish information leakage and retention channels: quantum information from earlier system-environment interactions is not immediately discarded but rather transferred to subsequent sub-environments. This retained information can then influence future collisions, breaking the memoryless property characteristic of Markovian dynamics and introducing temporal correlations in the system's evolution.

Consider the same setup as in [section 5.2](#): an  $n$ -qubit system with Hamiltonian  $H_S$  interacting with  $m$  sub-environments, each with Hamiltonian  $H_{E_j}$  and interaction Hamiltonian  $H_{I_j}$ . In the non-Markovian framework, each iteration consists of **two consecutive collisions**:

1. **System-environment collision:** System interacts with sub-environment  $j$  for time  $\Delta t$  via unitary  $U_j = e^{-i\lambda_j \bar{H}_j \Delta t}$
2. **Environment-environment collision:** Sub-environment  $j$  interacts with the next sub-environment  $j+1$  via a CPTP channel  $\mathcal{C}_{j,j+1}$
3. **Trace out:** Sub-environment  $j$  is traced out (after interacting with  $j+1$ )

This sequence is repeated for  $K$  iterations, where each iteration involves system-environment interaction followed by environment-environment interaction.

A natural choice for the inter-environment interaction is the partial swap operation, which interpolates between no information transfer and complete information exchange. For parameter  $p \in [0, 1]$  and two states  $\rho_j$  and  $\sigma_{j+1}$  of the same dimension:

$$\mathcal{C}_{j,j+1}[\rho_j \otimes \sigma_{j+1}] = (1-p)(\rho_j \otimes \sigma_{j+1}) + p \cdot S_{j,j+1}(\rho_j \otimes \sigma_{j+1})S_{j,j+1}^\dagger, \quad (5.4.1)$$

where  $S_{j,j+1}$  is the swap operator that exchanges the two states.

The parameter  $p$  controls the degree of non-Markovianity:

- $p = 0$ : No swap, reduces to memoryless (Markovian) collisions
- $p \in (0, 1)$ : Partial information transfer, non-Markovian dynamics
- $p = 1$ : Complete swap, maximal memory effects

Ciccarello et al. [74] showed that this partial swap mechanism, in the continuum limit, produces non-Markovian master equations with memory kernels.

To implement non-Markovian dynamics on a quantum computer, we need two environment registers (rather than one as in the Markovian case) that alternate roles across iterations.

Define the system-environment collision unitary:

$$U_{S_j} = \bar{U}_j \otimes I_E, \quad (5.4.2)$$

where  $\bar{U}_j = e^{-i\Delta t \lambda_j \bar{H}_j}$  acts on the system and one environment register, while  $I_E$  acts on the other environment register.

Define the environment-environment CPTP map:

$$\mathcal{V}_{j,j+1} = I_S \otimes \mathcal{C}_{j,j+1}, \quad (5.4.3)$$

which implements the partial swap on the two environment registers while leaving the system unchanged.

The  $j$ -th non-Markovian collision map  $\Phi_j^N$  depends on whether  $j$  is odd or even (determining which environment register stores the new sub-environment state):

**Definition 5.4.1** (Non-Markovian Collision Map). *For  $j \in [1, K - 1]$ , the  $j$ -th non-Markovian collision map is:*

$$\Phi_j^N[\cdot] = \begin{cases} \text{Tr}_{E_j}[\mathcal{V}_{j,j+1}[U_{S_j}(\cdot \otimes \rho_{E_{j+1}})U_{S_j}^\dagger]], & j \text{ odd} \\ \text{Tr}_{E_j}[\mathcal{V}_{j,j+1}[U_{S_j}(\cdot \otimes \rho_{E_{j+1}})U_{S_j}^\dagger]], & j \text{ even} \end{cases} \quad (5.4.4)$$

where the tensor product structure indicates which environment register holds the new sub-environment state  $\rho_{E_{j+1}}$ .

The key point is that after the  $j$ -th iteration, one environment register holds the state that will interact with the system in iteration  $j + 1$ , carrying forward memory of previous collisions. The full non-Markovian  $K$ -collision map composes  $K - 1$  of these collision maps, followed by a final system-environment collision (without subsequent environment-environment interaction):

$$\mathcal{N}_K[\cdot] \equiv \text{Tr}_{E_K} \left[ U_{S_K} \left( \bigcirc_{j=1}^{K-1} \Phi_j^{\mathcal{N}}[\cdot] \right) U_{S_K}^\dagger \right]. \quad (5.4.5)$$

This map describes the cumulative effect of  $K$  iterations of system-environment and environment-environment interactions.

As in the Markovian case, we approximate each system-environment unitary  $U_{S_j}$  using Hamiltonian simulation, replacing it with  $\tilde{U}_{S_j}$  satisfying  $\|U_{S_j} - \tilde{U}_{S_j}\| \leq \xi_j$ . The partial swap can be implemented exactly. The approximate non-Markovian collision maps  $\tilde{\Phi}_j^{\mathcal{N}}$  and the approximate  $K$ -collision map  $\tilde{\mathcal{N}}_K$  are defined analogously to their exact counterparts in Definition 5.4.1 and Equation 5.4.5, with  $\tilde{U}_{S_j}$  replacing  $U_{S_j}$ .

The following lemma establishes that the precision requirements for non-Markovian collision maps match those of the Markovian case:

**Lemma 5.4.2** (Bounds on Non-Markovian Approximate Collision Map). *Let  $O$  be an observable and  $\tilde{\mathcal{N}}_K$  be the approximate non-Markovian  $K$ -collision map, where*

$$\max_{1 \leq j \leq K} \|U_j - \tilde{U}_j\| \leq \frac{\varepsilon}{3K\|O\|}. \quad (5.4.6)$$

*Then, the expectation value of  $O$  with respect to the approximate map satisfies:*

$$|\text{Tr}[O \mathcal{N}_K[\rho]] - \text{Tr}[O \tilde{\mathcal{N}}_K[\rho]]| \leq \varepsilon. \quad (5.4.7)$$

*Proof.* The proof follows the same structure as Lemma 5.2.3, using the contractivity of CPTP maps and composition bounds. See [9] for details.  $\square$

## 5.4.2 Algorithm for Non-Markovian Dynamics

---

### Algorithm 6 Estimate Observable After Non-Markovian $K$ -Collision Map

---

**Require:** Initial system state  $\rho_S$ , sub-environment states  $\{\rho_{E_j}\}_{j=1}^K$ , observable  $O$ , unitaries  $\{\tilde{U}_j\}_{j=1}^K$  with precision  $\varepsilon/(3K\|O\|)$ , swap probability  $p$ , confidence  $\delta$

- 1: Initialize system in  $\rho_S$ , first environment register in  $\rho_{E_1}$ , ancilla in  $|+\rangle$
- 2: **for**  $j = 1$  to  $K - 1$  **do**
- 3:     Draw i.i.d. samples  $X_j, Y_j$  from LCU ensemble for  $\tilde{U}_j$
- 4:     Apply  $X_j^{(c)}$  and  $Y_j^{(a)}$  to system and appropriate environment register
- 5:     **if**  $j$  is odd **then**
- 6:         Initialize second environment register in  $\rho_{E_{j+1}}$
- 7:     **else**
- 8:         Initialize first environment register in  $\rho_{E_{j+1}}$
- 9:     **end if**
- 10:    With probability  $p$ , apply swap  $S_{j,j+1}$  between environment registers
- 11:    Trace out appropriate environment register
- 12: **end for**
- 13: Draw samples  $X_K, Y_K$  for final collision
- 14: Apply  $X_K^{(c)}$  and  $Y_K^{(a)}$  to system and remaining environment register
- 15: Trace out environment register
- 16: Measure ancilla-system on  $\sigma^x \otimes O$ , record outcome  $\mu_i$
- 17: Repeat for  $T = \mathcal{O}(\|O\|^2 \log(1/\delta)/\varepsilon^2)$  times
- 18: Return  $\mu = \frac{\xi^2}{T} \sum_{i=1}^T \mu_i$

**Ensure:**  $|\mu - \text{Tr}[O \mathcal{N}_K[\rho_S]]| \leq \varepsilon$  with probability  $\geq 1 - \delta$

---

The quantum circuit corresponding to each run of Algorithm 6 is shown in Figure 5.2. The circuit implements the non-Markovian  $K$ -collision map using Hamiltonian simulation by SA-LCU, with controlled and anti-controlled

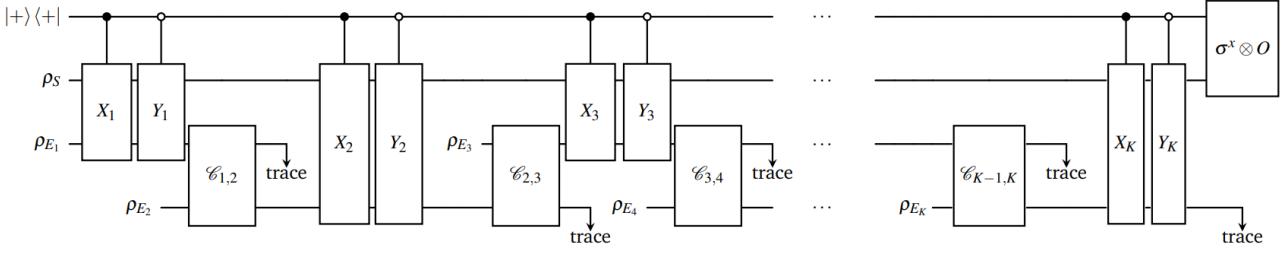


Figure 5.2: Quantum circuit corresponding to each run of Algorithm 6, simulating a non-Markovian  $K$ -collision map using Hamiltonian simulation by SA-LCU. The algorithm applies controlled and anti-controlled sampled unitaries ( $X_j, Y_j$ ) for the interaction between the system and each sub-environment, followed by an interaction between consecutive sub-environments using the channel ( $C_{j,j+1}$ ). This sequence is repeated for  $K$  collisions. At the end of the process, the ancilla qubit and the system are measured.

sampled unitaries ( $X_j, Y_j$ ) for system-environment interactions, followed by inter-environment channels ( $C_{j,j+1}$ ) between consecutive sub-environments.

The non-Markovian algorithm requires  $n$  system qubits, 2 environment registers (alternately reused), and 1 ancilla qubit, for a total qubit count of  $n + 2 \cdot n_{\text{env}} + 1$ , where  $n_{\text{env}}$  is the dimension of each sub-environment (typically 1 for single-qubit environments). The circuit depth matches the Markovian case asymptotically:

$$\tau_d^{\text{NM}} = \mathcal{O} \left( \lambda^2 K^2 \Delta t^2 \frac{\log(\lambda K \|O\| \Delta t / \varepsilon)}{\log \log(\lambda K \|O\| \Delta t / \varepsilon)} + K(\tau_{\rho_E} + \tau_{\text{swap}}) \right), \quad (5.4.8)$$

where  $\tau_{\text{swap}} = \mathcal{O}(n_{\text{env}})$  is the depth for implementing the swap gate.

The non-Markovian collision model framework enables simulation of a broad range of open system dynamics beyond Markovian Lindbladian evolution, including time-correlated noise processes, quantum systems with structured environments, memory-kernel master equations, and non-Markovian thermodynamic processes. By tuning the swap probability  $p$ , one can interpolate between fully Markovian ( $p = 0$ ) and strongly non-Markovian ( $p \rightarrow 1$ ) regimes, enabling systematic studies of memory effects in quantum dynamics. This extension demonstrates the flexibility and power of the collision model framework for simulating diverse open quantum system behaviors on early fault-tolerant quantum computers.

## 5.5 Simulating Lindbladian Dynamics via Collision Models

Lindbladian dynamics, described by the Gorini-Kossakowski-Sudarshan-Lindblad (GKSL) master equation, is a special case of Markovian dynamics that can be naturally realized through quantum collision models. In this section, we show how the Markovian  $K$ -collision map framework can be specialized to simulate Lindbladian evolution, providing an end-to-end quantum algorithm without requiring block encodings.

The most general form of a Markovian open quantum system evolution is given by the Lindblad master equation:

$$\frac{d\rho_S}{dt} = -i[H_S, \rho_S] + \sum_{k=1}^N \left( A_k \rho_S A_k^\dagger - \frac{1}{2} \{A_k^\dagger A_k, \rho_S\} \right), \quad (5.5.1)$$

where  $H_S$  generates coherent Hamiltonian evolution, and the Lindblad jump operators  $\{A_k\}_{k=1}^N$  capture dissipative effects from environmental coupling. The formal solution  $\rho_S(t) = e^{\mathcal{L}t}[\rho_S(0)]$  defines the Lindbladian map with superoperator  $\mathcal{L}$ . The key insight connecting Lindbladian dynamics to collision models is that repeated interactions with identically prepared sub-environments, in the limit of many weak collisions, reproduce Lindbladian evolution. This connection was rigorously established in Ref. [75]. To simulate Lindbladian evolution over time  $t$ , we use  $m$  sub-environments (typically equal to the system size) with  $K = m\nu$  total collisions. The discretization parameter  $\nu$  controls the collision time  $\Delta t = t/\nu$ , giving total evolution time  $K\Delta t = mt$ . The coupling strength must satisfy  $\lambda = \mathcal{O}(\nu/t)$  to ensure convergence to the Lindbladian. The approximation error between the  $(m, \nu)$ -collision map  $\mathcal{M}_{m,\nu}$  and the exact Lindblad map  $e^{\mathcal{L}t}$  is bounded by:

$$\|\mathcal{M}_{m,\nu} - e^{\mathcal{L}t}\|_{1 \rightarrow 1} = \mathcal{O} \left( \frac{t^2 m \Gamma}{\nu} \right), \quad (5.5.2)$$

where  $\Gamma = \sum_{k=1}^N \|A_k\|^2$  is the total dissipation strength. To achieve  $\varepsilon$ -approximation, we choose:

$$\nu = \mathcal{O}\left(\frac{t^2 m \Gamma}{\varepsilon}\right). \quad (5.5.3)$$

To simulate Lindbladian dynamics, we construct collision Hamiltonians  $\{H_\ell\}_{\ell=1}^m$  from the system Hamiltonian  $H_S$  and Lindblad operators  $\{A_k\}_{k=1}^N$  following the construction in Ref. [75]. The discretization parameter is set to  $\nu = \mathcal{O}(t^2 m \Gamma \|O\|/\varepsilon)$  to balance the collision model approximation error with Hamiltonian simulation error, yielding  $K = m\nu$  total collisions with collision time  $\Delta t = t/\nu$ . We then apply Algorithm 5 with these collision Hamiltonians, cycling through the  $m$  sub-environments repeatedly, to estimate  $\text{Tr}[O \cdot e^{\mathcal{L}t}[\rho_S]]$  to  $\varepsilon$ -accuracy with probability  $\geq 1 - \delta$ . It is worth noting that state-of-the-art direct approaches for Lindbladian simulation [73, 76, 77, 78] achieve query complexity  $\mathcal{O}(t \cdot \text{polylog}(t/\varepsilon))$ , which is near-optimal. However, these methods require block encodings of  $H_S$  and each Lindblad operator  $A_k$ , multiple ancilla qubits (typically  $\mathcal{O}(\log L)$  where  $L$  is the number of terms), and sophisticated controlled operations unsuitable for near-term devices. The collision model approach trades some efficiency—circuit depth  $\tilde{\mathcal{O}}(t^3/\varepsilon)$  versus  $\tilde{\mathcal{O}}(t/\varepsilon)$  for direct methods—for practical implementability on early fault-tolerant quantum computers without specialized oracles. Moreover, it provides a unified framework for comparing different Hamiltonian simulation methods, which we explore in the next section.

## 5.6 Numerical benchmarking

The collision model framework provides a unified platform for comparing different near-term Hamiltonian simulation techniques. In this section, we systematically analyze the circuit depth, qubit count, and classical repetitions required when using Trotterization (various orders), qDRIFT, SA-LCU, and qubitization as subroutines for simulating Lindbladian dynamics via the  $(m, \nu)$ -collision map.

For simulating Lindblad dynamics with evolution time  $t$  to precision  $\varepsilon$  in the observable  $O$ , the collision model uses  $K = m\nu$  total collisions with discretization parameter  $\nu = \mathcal{O}(t^2 m \Gamma \|O\|/\varepsilon)$ , giving time per collision  $\Delta t = t/\nu$ . The collision Hamiltonians have maximum normalization factor  $\lambda_{\max} = \max_\ell \lambda_\ell$  and at most  $L = \max_\ell L_\ell$  Pauli terms. Each Hamiltonian simulation subroutine must achieve precision  $\varepsilon' = \varepsilon/(3K\|O\|)$  to ensure overall observable estimation error  $\leq \varepsilon$ .

Method	Circuit Depth	Qubits	Repetitions
First-order Trotter	$\tilde{\mathcal{O}}\left(\frac{Lm^3 t^3 \ O\ ^2 \Gamma \lambda_{\max}^2}{\varepsilon^2}\right)$	$n + 1$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
Second-order Trotter	$\tilde{\mathcal{O}}\left(L(m t)^{9/4} \left(\frac{\ O\ }{\varepsilon}\right)^{5/4} (\Gamma \lambda_{\max}^2)^{3/4}\right)$	$n + 1$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
High-order Trotter	$\tilde{\mathcal{O}}\left(\frac{Lm^2 t^2 \Gamma \lambda_{\max} \ O\ }{\varepsilon}\right)$	$n + 1$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
qDRIFT	$\tilde{\mathcal{O}}\left(\frac{m^3 t^3 \Gamma \lambda_{\max}^2 \ O\ ^2}{\varepsilon^2}\right)$	$n + 1$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
SA-LCU	$\tilde{\mathcal{O}}\left(\frac{m^3 t^3 \Gamma \lambda_{\max}^2 \ O\ }{\varepsilon}\right)$	$n + 2$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$
Qubitization	$\tilde{\mathcal{O}}\left(\frac{Lm^2 t^2 \Gamma \lambda_{\max} \ O\ }{\varepsilon}\right)$	$n + 1 + \mathcal{O}(\log L)$	$\mathcal{O}\left(\frac{\ O\ ^2}{\varepsilon^2}\right)$

Table 5.1: Comparison of circuit depths for Lindbladian simulation via collision models. All expressions suppress polylogarithmic factors and constants. Parameters:  $m$  (system size),  $t$  (evolution time),  $\varepsilon$  (precision),  $\|O\|$  (observable norm),  $\Gamma$  (dissipation strength),  $\lambda_{\max}$  (max Hamiltonian norm),  $L$  (max number of terms).

To concretely compare the performance of different Hamiltonian simulation methods for collision model-based Lindbladian simulation, we apply our algorithms to a physically relevant benchmark: a one-dimensional transverse-field Ising model (Heisenberg XXX model) undergoing amplitude damping. The corresponding Lindblad master equation for this system is:

$$\frac{d\rho_S}{dt} = -i[H_S/m, \rho_S] + \sum_{j=1}^m \left( \sqrt{\gamma} \sigma_j^- \rho_S \sigma_j^+ - \frac{1}{2} \{\sigma_j^+, \sigma_j^-, \rho_S\} \right), \quad (5.6.1)$$

where the system Hamiltonian is rescaled by  $m$  to account for sequential interactions, and the Lindblad jump operators are  $A_j = \sqrt{\gamma}\sigma_j^-$  for each site. The total dissipation strength is  $\Gamma = \sum_{j=1}^m \|A_j\|^2 = m\gamma = 1.0$ . We consider a 10-qubit ( $m = 10$ ) Ising chain with nearest-neighbor interactions and system Hamiltonian

$$H_S = -J \sum_{i=1}^{m-1} \sigma_i^z \sigma_{i+1}^z - h \sum_{i=1}^m \sigma_i^x, \quad (5.6.2)$$

where  $J$  is the coupling strength between nearest neighbors,  $h$  is the transverse magnetic field strength, and  $\sigma_i^z, \sigma_i^x$  are Pauli operators on site  $i$ . For our benchmarks, we set  $J = h = 1$ , giving  $\lambda_S = (J + h) = 2$ . The environment consists of  $m = 10$  single-qubit sub-environments, each initially in the ground state  $|0\rangle$  (corresponding to zero temperature  $\omega \rightarrow \infty$ ) with

$$\rho_{E_j} = |0\rangle\langle 0|, \quad \forall j \in [1, m], \quad (5.6.3)$$

and sub-environment Hamiltonian  $H_{E_j} = \omega|1\rangle\langle 1|$  (number operator). The interaction between the system and the  $j$ -th sub-environment implements amplitude damping, an energy dissipation process where excitations decay to the ground state, via

$$H_{I_j} = \sqrt{\gamma} \left( \mathbb{I}^{\otimes(j-1)} \otimes \sigma_j^+ \otimes \mathbb{I}^{\otimes(m-j)} \otimes \sigma_a^- + \text{h.c.} \right), \quad (5.6.4)$$

where  $\sigma^\pm = (\sigma^x \pm i\sigma^y)/2$  are the raising/lowering operators, also called ladder operators, which satisfy  $\sigma^+ = |1\rangle\langle 0|$  and  $\sigma^- = |0\rangle\langle 1|$ , while  $\gamma$  is the damping strength, subscript  $a$  denotes the environment qubit, and h.c. denotes Hermitian conjugate. We set  $\gamma = 0.1$  for our benchmarks.

We estimate the average transverse-field magnetization:

$$M_z = \frac{1}{m} \sum_{j=1}^m \sigma_j^z, \quad (5.6.5)$$

with respect to the reduced state  $e^{\mathcal{L}t}[\rho_S]$  after Lindbladian evolution. That is, we compute  $\mu$  such that:

$$|\mu - \text{Tr}[M_z \cdot e^{\mathcal{L}t}[\rho_S]]| \leq \varepsilon. \quad (5.6.6)$$

Note that  $\|M_z\| = 1$  since it is a normalized average of single-qubit observables.

For the  $(m, \nu)$ -collision map approximation with  $m = 10$ , we use total collisions  $K = m\nu = 10\nu$ , discretization  $\nu = \mathcal{O}(t^2 m \Gamma \|M_z\|/\varepsilon) = \mathcal{O}(10t^2/\varepsilon)$ , collision time  $\Delta t = t/\nu$ , and coupling strength  $\lambda = \mathcal{O}(\nu/t)$  to ensure the Lindbladian limit. Each collision Hamiltonian  $H_j = H_S/m + H_{E_j} + H_{I_j}$  has approximately  $L_j \approx 2m + 3 = 23$  terms (rescaled Ising + interaction + environment) and normalization factor  $\lambda_j \approx \lambda_S/m + \sqrt{\gamma\lambda} \approx 0.2 + \sqrt{0.1 \cdot \nu/t}$ .

We now examine how the system evolves under different environment energy levels. Figure 5.3 shows the magnetization evolution for various environment Hamiltonian coefficients  $\omega$ . Higher values of  $\omega$  (larger energy gap in the environment) lead to faster decay toward the steady state  $|0\rangle\langle 0|$ , which has magnetization  $M_z = 1$  (fully aligned). This demonstrates the tunability of dissipation rates through the environment energy parameter.

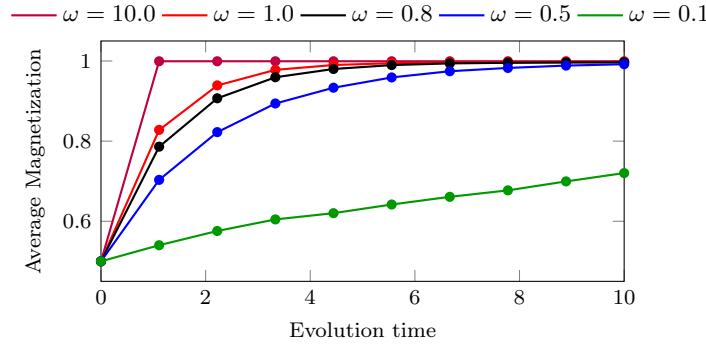


Figure 5.3: Collision model dynamics for different environment energy levels  $\omega$  (coefficient of the environment Hamiltonian  $H_{E_j} = \omega|1\rangle\langle 1|$ ). Higher values of  $\omega$  result in faster decay to the steady state with magnetization  $M_z = 1$ , demonstrating how the environment energy parameter controls the dissipation rate.

To measure the computational cost of different Hamiltonian simulation methods, we fix evolution time  $t = 1$  and vary precision  $\varepsilon \in [10^{-2}, 10^{-5}]$ . For each method, we compute the CNOT gate count per coherent run of the algorithm. Figure 5.4 (Left) shows CNOT counts as a function of precision  $\varepsilon$  on a log-log scale.

The results reveal significant differences in performance across methods. At high precision ( $\varepsilon = 10^{-4}$ ), SA-LCU requires approximately  $5 \times 10^5$  CNOTs, while second-order Trotter needs  $1 \times 10^8$  CNOTs (200 times more), qDRIFT requires  $1 \times 10^9$  CNOTs (2000 times more), and first-order Trotter demands  $2 \times 10^9$  CNOTs (4000 times more than SA-LCU). The log-log plot slopes confirm our theoretical predictions: SA-LCU exhibits a slope of approximately -1, indicating linear scaling in  $1/\varepsilon$ ; second-order Trotter shows a slope around -1.25, corresponding to  $1/\varepsilon^{1.25}$  scaling; while both qDRIFT and first-order Trotter display slopes near -2, confirming their quadratic dependence on  $1/\varepsilon$ . These results demonstrate SA-LCU's superior performance for high-precision applications.

Next, we fix precision  $\varepsilon = 0.01$  and vary evolution time  $t \in [10, 100]$ . Again, we compute CNOT gate counts per coherent run. Figure 5.4 (Right) shows CNOT counts as a function of time  $t$  on a log-log scale for a 10-qubit Ising model. The time-scaling benchmark shows different relative performance across methods. At moderate evolution time ( $t = 10$ ), SA-LCU requires approximately  $8 \times 10^7$  CNOTs, second-order Trotter needs  $3 \times 10^8$  CNOTs (4 times more), first-order Trotter requires  $4 \times 10^{10}$  CNOTs (500 times more), and qDRIFT demands  $5 \times 10^{12}$  CNOTs (70,000 times more than SA-LCU). However, the situation changes for long evolution times. At  $t = 100$ , second-order Trotter becomes the most efficient method, requiring only  $5 \times 10^{10}$  CNOTs, while SA-LCU needs  $6 \times 10^{11}$  CNOTs (11 times more than second-order Trotter), and first-order Trotter and qDRIFT demand  $4 \times 10^{13}$  and  $1 \times 10^{18}$  CNOTs respectively. The crossover point where second-order Trotter becomes more efficient than SA-LCU occurs around  $t \approx 20$ .

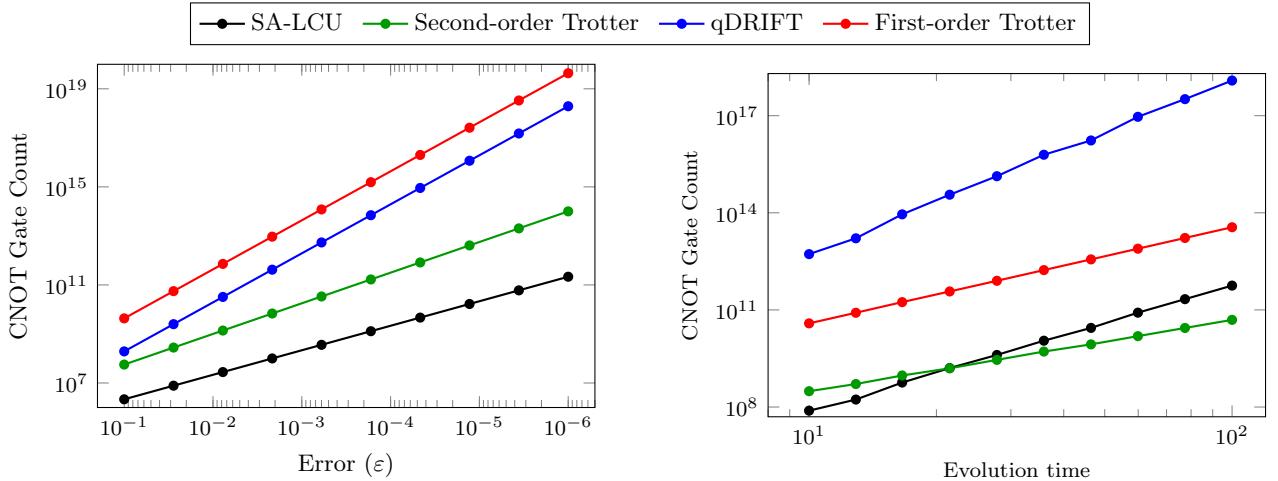


Figure 5.4: CNOT gate count benchmarks for collision model Lindbladian simulation. (Left) Gate count vs. precision  $\varepsilon$  for fixed evolution time  $t = 1$ . (Right) Gate count vs. evolution time for fixed precision  $\varepsilon = 0.01$ .

The collision model framework naturally extends to non-Markovian dynamics through inter-environment interactions, typically implemented via partial swap operations between successive sub-environments. Crucially, the transition from Markovian to non-Markovian dynamics only modifies the environment-environment coupling structure; the system-environment collision unitaries  $U_j = e^{-i\lambda_j \bar{H}_j \Delta t}$  remain identical to those in the Markovian case. Since the Hamiltonian simulation methods benchmarked above are applied exclusively to these collision unitaries, the relative performance rankings and scaling behaviors established in the preceding analysis apply directly to non-Markovian simulations. Consequently, we do not present separate Hamiltonian simulation benchmarks for non-Markovian dynamics; instead, we demonstrate the physical effects of non-Markovianity on system evolution. Figure 5.5 shows how the partial swap parameter  $p$  controls the degree of non-Markovianity in a 6-qubit Ising system. As  $p$  increases from 0 to 1, inter-environment exchange transfers information between successive sub-environments, preventing them from remaining in the pure  $|0\rangle$  state. This mixing weakens the effective amplitude damping channel: low  $p$  values ( $p = 0, p = 0.9$ ) produce strong damping with rapid magnetization saturation, while high  $p$  values ( $p = 0.999, p = 1$ ) yield weaker damping with slower, reduced saturation.

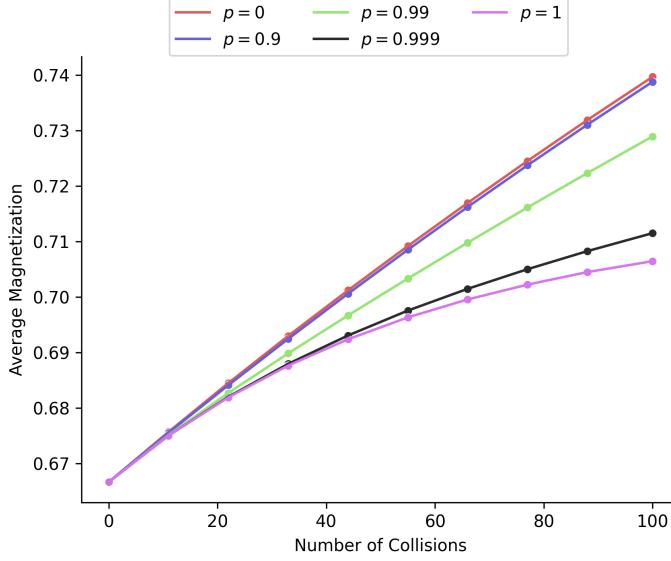


Figure 5.5: Effect of partial swap parameter  $p$  on magnetization evolution in a 6-qubit Ising chain under non-Markovian collision model dynamics.

All gate counts in our benchmarks are computed under standard quantum circuit implementation assumptions: Pauli rotations  $e^{-i\theta P}$  are implemented with  $\mathcal{O}(n)$  CNOT gates, Clifford gates have unit cost, and environmental state preparation in computational basis states incurs no gate overhead. The partial swap operations for non-Markovian dynamics add  $\mathcal{O}(n_{\text{env}})$  gates independent of the Hamiltonian simulation method, preserving the relative performance rankings across all methods.

The numerical benchmarks demonstrate regime-dependent performance across Hamiltonian simulation techniques. In the high-precision regime ( $\varepsilon \sim 10^{-4}$ ,  $t = 1$ ), SA-LCU achieves  $200\text{-}4000\times$  fewer gates than competing methods, with observed linear scaling in  $1/\varepsilon$  compared to quadratic scaling for first-order Trotter and qDRIFT. Second-order Trotter exhibits intermediate  $\varepsilon^{-1.25}$  scaling. In the time-scaling regime ( $\varepsilon = 0.01$ ), SA-LCU maintains superior performance for short-to-moderate evolution times ( $t \lesssim 3$ ), but second-order Trotter becomes more efficient for long evolution times ( $t \gtrsim 3$ ) due to its  $t^{2.25}$  scaling compared to cubic scaling for other methods. The extension to non-Markovian dynamics through inter-environment interactions demonstrates that collision models can simulate both Markovian and memory-retaining open system dynamics using these Hamiltonian simulation techniques.

# Chapter 6

## Conclusion

Early fault-tolerant quantum computers represent a critical intermediate stage between today’s noisy intermediate-scale quantum devices and future fully fault-tolerant quantum computers [18]. This transition period is characterized by partial error correction capabilities that enable the execution of algorithms requiring moderate circuit depths, specifically those algorithms that NISQ cannot reliably execute but that do not yet demand the extensive resources of full fault tolerance. These devices possess a limited number of physical qubits, imposing constraints on the distance of the error-correcting codes they can support, deviating from the conventional assumptions of fault-tolerant quantum computing where such resources are presumed to be infinitely scalable. This finite resource constraint creates a unique computational regime where algorithms such as Quantum Phase Estimation, which require many coherent quantum operations in sequence beyond NISQ capabilities, become feasible. The single ancilla LCU algorithm investigated in this thesis [9] represents the type of resource-efficient approach suited to this regime. By performing LCU operations with minimal ancilla overhead, a critical consideration when logical qubits are scarce, this method aligns with the constraints of early fault-tolerant devices. However, the trade-off is significant: the algorithm cannot serve as a general subroutine across different problem domains, nor can it leverage amplitude amplification to reduce the number of required classical runs, necessitating careful analysis of when this approach provides tangible benefit.

The early fault-tolerant regime fundamentally demands a shift from asymptotic complexity analysis to concrete numerical benchmarking. Rather than focusing on asymptotic behavior, researchers must consider the overhead savings that may be possible in the scale of devices expected in the near term. This shift is not merely a preference but a necessity imposed by the fundamental constraints of these devices. The key observation is that early fault-tolerant computing operates in the pre-asymptotic regime, where constant factors dominate algorithmic performance. While asymptotic scaling may appear favorable, in practice the constant factors hidden by big-O notation can be quite large, ranging from hundreds to thousands for efficient protocols to much larger values for protocols optimized for high threshold. An algorithm with better asymptotic complexity but a large constant factor can perform substantially worse than a theoretically inferior algorithm at practically achievable scales. Beyond constant factors, non-asymptotic optimizations such as tight commutator bounds can fundamentally alter algorithm selection [79]. For sparse Hamiltonians with favorable commutator structure, product formulas leveraging these bounds can outperform methods with superior asymptotic scaling, demonstrating that problem-specific structure determines efficiency more than abstract complexity parameters. The numerical benchmarking performed in this thesis, comparing single ancilla LCU against ancilla-less Hamiltonian simulation methods using Truncated Taylor Series [80], and extending this analysis to groundstate estimation and collision model simulation, demonstrates this approach. Rather than claiming asymptotic superiority, this work provides concrete resource counts: exact gate requirements, qubit overhead, and circuit depth for specific problem instances. Such analysis enables direct assessment of whether single ancilla LCU provides concrete gains at achievable scales, accounting for the algorithm’s limitations against its advantages of reduced ancilla requirements.

While this thesis has established numerical benchmarks for single ancilla LCU in several applications, significant work remains to fully characterize its utility in the early fault-tolerant landscape. The benchmarking framework developed here can be naturally extended to other single ancilla LCU applications beyond Hamiltonian simulation, groundstate estimation, and collision models. Any quantum algorithm requiring LCU operations could potentially benefit from this resource-efficient approach, though each application domain requires careful numerical analysis to determine whether the trade-offs remain favorable. A crucial future direction involves comparative benchmarking against emerging early fault-tolerant algorithms such as THRIFT [55] and other phase estimation variants optimized for partial error correction, and the methodology developed here could

be directly transferable to analyzing their resource requirements. Similarly, while standard Quantum Singular Value Transformation requires extensive ancilla overhead and multi-qubit controlled operations suited only to full fault tolerance, recent variants that avoid block encodings achieve comparable functionality with minimal ancilla [19], opening QSVT-based methods to the early fault-tolerant regime. However, an important limitation lies in incorporating the complete computational cost stack. The current benchmarking focuses primarily on logical gate counts, qubit requirements, and circuit depth. Future work must integrate the overhead of encoding classical data into quantum states and reading out measurement results, which can be substantial for algorithms requiring many quantum-classical iterations. Benchmarking must also account for the full compilation pipeline from logical gates to hardware-native operations, where even optimized translation introduces significant overhead that requires careful compilation to minimize error accumulation. The design and performance of the classical control software that compiles algorithms and controls the quantum computer has a non-negligible impact on resources, and decoder throughput, latency, classical compute requirements, and memory footprint must be explicitly modeled, as decoder performance can become the actual bottleneck. Extending this work to model performance on realistic architectures, whether superconducting qubits with surface codes [14], neutral atoms with qLDPC codes [16], or trapped ions with different error characteristics, would provide actionable insights into when and where single ancilla LCU becomes practically advantageous. Incorporating correlated errors, crosstalk, and hardware-specific noise patterns would further strengthen the analysis beyond idealized error models. The benchmarking framework developed in this thesis provides a foundation for algorithm selection in the early fault-tolerant era, with future work extending to end-to-end resource estimation across the full computational stack.

# Bibliography

- [1] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6):467–488, Jun 1982. ISSN 1572-9575. doi:[10.1007/BF02650179](https://doi.org/10.1007/BF02650179). URL <https://doi.org/10.1007/BF02650179>.
- [2] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pages 212–219. ACM, 1996. doi:[10.1145/237814.237866](https://doi.org/10.1145/237814.237866).
- [3] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997. doi:[10.1137/S0097539795293172](https://doi.org/10.1137/S0097539795293172).
- [4] Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15):150502, 2009. doi:[10.1103/PhysRevLett.103.150502](https://doi.org/10.1103/PhysRevLett.103.150502).
- [5] Lin Lin and Yu Tong. Near-optimal ground state preparation. *Quantum*, 4:372, December 2020. ISSN 2521-327X. doi:[10.22331/q-2020-12-14-372](https://doi.org/10.22331/q-2020-12-14-372). URL <https://doi.org/10.22331/q-2020-12-14-372>.
- [6] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O’Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1):4213, 2014. doi:[10.1038/ncomms5213](https://doi.org/10.1038/ncomms5213).
- [7] Dominic W. Berry, Andrew M. Childs, Aaron Ostrander, and Guoming Wang. Quantum algorithm for linear differential equations with exponentially improved dependence on precision. *Communications in Mathematical Physics*, 356(3):1057–1081, Dec 2017. ISSN 1432-0916. doi:[10.1007/s00220-017-3002-y](https://doi.org/10.1007/s00220-017-3002-y). URL <https://doi.org/10.1007/s00220-017-3002-y>.
- [8] Kushagra Garg, Zeeshan Ahmed, Subhadip Mitra, and Shantanav Chakraborty. Simulating quantum collision models with Hamiltonian simulations using early fault-tolerant quantum computers. *Phys. Rev. A*, 112:022425, Aug 2025. doi:[10.1103/PhysRevA.112.022425](https://doi.org/10.1103/PhysRevA.112.022425). URL <https://journals.aps.org/prabstract/10.1103/PhysRevA.112.022425>.
- [9] Shantanav Chakraborty. Implementing any linear combination of unitaries on intermediate-term quantum computers. *Quantum*, 8:1496, October 2024. ISSN 2521-327X. doi:[10.22331/q-2024-10-10-1496](https://doi.org/10.22331/q-2024-10-10-1496). URL <https://doi.org/10.22331/q-2024-10-10-1496>.
- [10] Guang Hao Low and Isaac L. Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3:163, Jul 2019. ISSN 2521-327X. doi:[10.22331/q-2019-07-12-163](https://doi.org/10.22331/q-2019-07-12-163). URL <https://doi.org/10.22331/q-2019-07-12-163>.
- [11] Guang Hao Low and Isaac L. Chuang. Optimal hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118:010501, Jan 2017. doi:[10.1103/PhysRevLett.118.010501](https://doi.org/10.1103/PhysRevLett.118.010501). URL <https://doi.org/10.1103/PhysRevLett.118.010501>.
- [12] Frank Arute et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. doi:[10.1038/s41586-019-1666-5](https://doi.org/10.1038/s41586-019-1666-5).
- [13] Han-Sen Zhong et al. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020. doi:[10.1126/science.abe8770](https://doi.org/10.1126/science.abe8770).
- [14] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Physical Review A*, 86(3):032324, 2012. doi:[10.1103/PhysRevA.86.032324](https://doi.org/10.1103/PhysRevA.86.032324).

- [15] Nikolas P. Breuckmann and Jens Niklas Eberhardt. Quantum low-density parity-check codes. *PRX Quantum*, 2(4):040101, Oct 2021. doi:[10.1103/PRXQuantum.2.040101](https://doi.org/10.1103/PRXQuantum.2.040101).
- [16] Diego Ruiz, Jérémie Guillaud, Anthony Leverrier, Mazyar Mirrahimi, and Christophe Vuillot. LDPC-cat codes for low-overhead quantum computing in 2D. *Nature Communications*, 16(1):1040, 2025. doi:[10.1038/s41467-025-56298-8](https://doi.org/10.1038/s41467-025-56298-8).
- [17] Ben W. Reichardt, David Aasen, Rui Chao, Alex Chernoguzov, Wim van Dam, John P. Gaebler, Dan Gresh, Dominic Lucchetti, Michael Mills, Steven A. Moses, Brian Neyenhuis, Adam Paetznick, Andres Paz, Peter E. Siegfried, Marcus P. da Silva, Krysta M. Svore, Zhenghan Wang, and Matt Zanner. Demonstration of quantum computation and error correction with a tesseract code. *arXiv preprint arXiv:2409.04628*, 2024.
- [18] Amara Katabarwa, Katerina Gratsea, Athena Caesura, and Peter D. Johnson. Early fault-tolerant quantum computing. *PRX Quantum*, 5:020101, Jun 2024. doi:[10.1103/PRXQuantum.5.020101](https://doi.org/10.1103/PRXQuantum.5.020101).
- [19] Shantanav Chakraborty, Soumyabrata Hazra, Tongyang Li, Changpeng Shao, Xinzhou Wang, and Yuxin Zhang. Quantum singular value transformation without block encodings: Near-optimal complexity with minimal ancilla. *arXiv preprint arXiv:2504.02385*, 2025.
- [20] Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, Aug 1996. doi:[10.1126/science.273.5278.1073](https://doi.org/10.1126/science.273.5278.1073).
- [21] Masuo Suzuki. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Journal of Mathematical Physics*, 32(2):400–407, 1991. doi:[10.1063/1.529425](https://doi.org/10.1063/1.529425).
- [22] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu. A theory of trotter error with commutator scaling. *Physical Review X*, 11(1):011020, 2021. doi:[10.1103/PhysRevX.11.011020](https://doi.org/10.1103/PhysRevX.11.011020).
- [23] Sergiy Zhuk, Nathan Robertson, and Sergey Bravyi. Trotter error bounds and dynamic multi-product formulas for hamiltonian simulation. *Physical Review Research*, 6(3):033309, 2024. doi:[10.1103/PhysRevResearch.6.033309](https://doi.org/10.1103/PhysRevResearch.6.033309).
- [24] Earl T. Campbell. Random compiler for fast hamiltonian simulation. *Physical Review Letters*, 123(7):070503, 2019. doi:[10.1103/PhysRevLett.123.070503](https://doi.org/10.1103/PhysRevLett.123.070503).
- [25] Tunde David, Ilya Sinayskiy, and Francesco Petruccione. Tighter error bounds for the qdrift algorithm. *arXiv preprint arXiv:2506.17199*, 2025.
- [26] Kouhei Nakaji, Mohsen Bagherimehrab, and Alán Aspuru-Guzik. qswift: High-order randomized compiler for hamiltonian simulation. *PRX Quantum*, 5(2):020330, 2024. doi:[10.1103/PRXQuantum.5.020330](https://doi.org/10.1103/PRXQuantum.5.020330).
- [27] Minh C. Tran, Yuan Su, Daniel Carney, and Jacob M. Taylor. Faster digital quantum simulation by symmetry protection. *PRX Quantum*, 2(1):010323, 2021. doi:[10.1103/PRXQuantum.2.010323](https://doi.org/10.1103/PRXQuantum.2.010323).
- [28] Andrew M. Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Information and Computation*, 12(11-12):901–924, 2012. doi:[10.26421/QIC12.11-12-1](https://doi.org/10.26421/QIC12.11-12-1).
- [29] Jacob T Seeley, Martin J Richard, and Peter J Love. The bravyi-kitaev transformation for quantum computation of electronic structure. *The Journal of Chemical Physics*, 137(22):224109, 2012. doi:[10.1063/1.4768229](https://doi.org/10.1063/1.4768229).
- [30] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C Benjamin, and Xiao Yuan. Quantum computational chemistry. *Reviews of Modern Physics*, 92(1):015003, 2020. doi:[10.1103/RevModPhys.92.015003](https://doi.org/10.1103/RevModPhys.92.015003).
- [31] Yimin Ge, Jordi Tura, and J. Ignacio Cirac. Faster ground state preparation and high-precision ground energy estimation with fewer qubits. *Journal of Mathematical Physics*, 60(2):022202, 2019. doi:[10.1063/1.5027484](https://doi.org/10.1063/1.5027484). URL <https://doi.org/10.1063/1.5027484>.
- [32] Valerio Scarani, Mário Ziman, Peter Štelmachovič, Nicolas Gisin, and Vladimír Bužek. Thermalizing quantum machines: Dissipation and entanglement. *Phys. Rev. Lett.*, 88:097905, Feb 2002. doi:[10.1103/PhysRevLett.88.097905](https://doi.org/10.1103/PhysRevLett.88.097905).
- [33] Francesco Ciccarello, Salvatore Lorenzo, Vittorio Giovannetti, and G. Massimo Palma. Quantum collision models: Open system dynamics from repeated interactions. *Physics Reports*, 954:1–70, 2022. ISSN 0370-1573. doi:[10.1016/j.physrep.2022.01.001](https://doi.org/10.1016/j.physrep.2022.01.001).

- [34] John von Neumann. *Mathematical Foundations of Quantum Mechanics*. Princeton University Press, 1955. English translation by Robert T. Beyer; original German edition 1932.
- [35] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, Cambridge, UK, 2010.
- [36] Paul Benioff. The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of Statistical Physics*, 22(5):563–591, May 1980. ISSN 1572-9613. doi:[10.1007/BF01011339](https://doi.org/10.1007/BF01011339). URL <https://doi.org/10.1007/BF01011339>.
- [37] David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society A*, 400(1818):97–117, 1985. doi:[10.1098/rspa.1985.0070](https://doi.org/10.1098/rspa.1985.0070).
- [38] David Deutsch. Quantum computational networks. *Proceedings of the Royal Society A*, 425(1868):73–90, 1989. doi:[10.1098/rspa.1989.0099](https://doi.org/10.1098/rspa.1989.0099).
- [39] A. Yu. Kitaev. Quantum computations: algorithms and error correction. *Russian Mathematical Surveys*, 52(6):1191–1249, 1997. doi:[10.1070/RM1997v05n06ABEH002155](https://doi.org/10.1070/RM1997v05n06ABEH002155).
- [40] Christopher M. Dawson and Michael A. Nielsen. The solovay–kitaev algorithm. *Quantum Information & Computation*, 6(1):81–95, 2005. See also arXiv:quant-ph/0505030.
- [41] John S. Bell. On the Einstein Podolsky Rosen paradox. *Physics Physique Fizika*, 1(3):195–200, 1964. doi:[10.1103/PhysicsPhysiqueFizika.1.195](https://doi.org/10.1103/PhysicsPhysiqueFizika.1.195).
- [42] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A. Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52(5):3457–3467, 1995. doi:[10.1103/PhysRevA.52.3457](https://doi.org/10.1103/PhysRevA.52.3457).
- [43] Vadym Kliuchnikov, Alex Bocharov, Martin Roetteler, and Jon Yard. A framework for approximating qubit unitaries, 2015.
- [44] K. Kim, M.-S. Chang, S. Korenblit, R. Islam, E. E. Edwards, J. K. Freericks, G.-D. Lin, L.-M. Duan, and C. Monroe. Quantum simulation of the transverse ising model with trapped ions. *New Journal of Physics*, 13:105003, Oct 2011. doi:[10.1088/1367-2630/13/10/105003](https://doi.org/10.1088/1367-2630/13/10/105003).
- [45] Yudong Cao, Jonathan Romero, Jonathan P. Olson, Matthias Degroote, Peter D. Johnson, Mária Kieferová, Ian D. Kivlichan, Tim Menke, Borja Peropadre, Nicolas P. D. Sawaya, Sukin Sim, Libor Veis, and Alán Aspuru-Guzik. Quantum chemistry in the age of quantum computing. *Chemical Reviews*, 119(19):10856–10915, 2019. doi:[10.1021/acs.chemrev.8b00803](https://doi.org/10.1021/acs.chemrev.8b00803).
- [46] J. Hubbard. Electron correlations in narrow energy bands. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 276(1365):238–257, 1963. doi:[10.1098/rspa.1963.0204](https://doi.org/10.1098/rspa.1963.0204).
- [47] Assa Auerbach. *Interacting Electrons and Quantum Magnetism*. Springer-Verlag, New York, 1994. doi:[10.1007/978-1-4612-0869-3](https://doi.org/10.1007/978-1-4612-0869-3).
- [48] A. Yu. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and Quantum Computation*, volume 47 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2002. doi:[10.1090/gsm/047](https://doi.org/10.1090/gsm/047).
- [49] Brian C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*, volume 222 of *Graduate Texts in Mathematics*. Springer, 2nd edition, 2015. doi:[10.1007/978-3-319-13467-3](https://doi.org/10.1007/978-3-319-13467-3).
- [50] Dominic W. Berry, Andrew M. Childs, and Robin Kothari. Hamiltonian simulation with nearly optimal dependence on all parameters. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 792–809, 2015. doi:[10.1109/FOCS.2015.54](https://doi.org/10.1109/FOCS.2015.54).
- [51] C. Monroe, W. C. Campbell, L.-M. Duan, Z.-X. Gong, A. V. Gorshkov, P. W. Hess, R. Islam, K. Kim, N. M. Linke, G. Pagano, P. Richerme, C. Senko, and N. Y. Yao. Programmable quantum simulations of spin systems with trapped ions. *Reviews of Modern Physics*, 93(2):025001, Apr 2021. doi:[10.1103/RevModPhys.93.025001](https://doi.org/10.1103/RevModPhys.93.025001).
- [52] Yingkai Ouyang, David R. White, and Earl T. Campbell. Compilation by stochastic Hamiltonian sparsification. *Quantum*, 4:235, Feb 2020. doi:[10.22331/q-2020-02-27-235](https://doi.org/10.22331/q-2020-02-27-235).

- [53] Matthew Hagan and Nathan Wiebe. Composite quantum simulations. *Quantum*, 7:1181, Nov 2023. doi:[10.22331/q-2023-11-14-1181](https://doi.org/10.22331/q-2023-11-14-1181).
- [54] Matthew Pocanic, Matthew Hagan, Juan Carrasquilla, Dvira Segal, and Nathan Wiebe. Composite Qdrift-product formulas for quantum and classical simulations in real and imaginary time. *Physical Review Research*, 6(1):013224, Mar 2024. doi:[10.1103/PhysRevResearch.6.013224](https://doi.org/10.1103/PhysRevResearch.6.013224).
- [55] J. L. Bosse, A. M. Childs, C. Derby, F. M. Gambetta, H. Krovi, B. K. Lee, T. Peng, R. D. Somma, M. Sahinoglu, D. Barberena, G. S. Barron, D. Camps, A. Chan, D. J. Gorman, L. C. G. Govia, C. Hadfield, N. Ingraham, I. Hincks, J. Marshall, S. T. Merkel, M. E. S. Morales, M. A. Perlin, J. Schuyler, R. Shaydulin, A. Schmitz, E. Shumpert, N. Sridhar, B. Swingle, E. Tyhurst, R. Van Beeumen, N. Wiebe, F. Zhang, and L. Zhu. Efficient and practical hamiltonian simulation from time-dependent product formulas. *Nature Communications*, 16:2673, Mar 2025. doi:[10.1038/s41467-025-57580-5](https://doi.org/10.1038/s41467-025-57580-5). URL <https://doi.org/10.1038/s41467-025-57580-5>.
- [56] Guang Hao Low and Nathan Wiebe. Hamiltonian simulation in the interaction picture. *arXiv preprint*, 2018. doi:[10.48550/arXiv.1805.00675](https://doi.org/10.48550/arXiv.1805.00675).
- [57] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Simulating hamiltonian dynamics with a truncated taylor series. *Physical Review Letters*, 114(9):090502, 2015. doi:[10.1103/PhysRevLett.114.090502](https://doi.org/10.1103/PhysRevLett.114.090502).
- [58] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 193–204, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367059. doi:[10.1145/3313276.3316366](https://doi.org/10.1145/3313276.3316366). URL <https://doi.org/10.1145/3313276.3316366>.
- [59] Andrew M Childs, Dmitri Maslov, Yunseong Nam, Neil J Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, 2018. doi:[10.1073/pnas.1801723115](https://doi.org/10.1073/pnas.1801723115).
- [60] Kianna Wan, Mario Berta, and Earl T. Campbell. Randomized quantum algorithm for statistical phase estimation. *Phys. Rev. Lett.*, 129:030503, Jul 2022. doi:[10.1103/PhysRevLett.129.030503](https://doi.org/10.1103/PhysRevLett.129.030503).
- [61] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu. Theory of trotter error with commutator scaling. *Physical Review X*, 11(1), Jan 2021. ISSN 2160-3308. doi:[10.1103/physrevx.11.011020](https://doi.org/10.1103/physrevx.11.011020). URL <http://dx.doi.org/10.1103/PhysRevX.11.011020>.
- [62] Earl T. Campbell. Random compiler for fast hamiltonian simulation. *Phys. Rev. Lett.*, 123:070503, Aug 2019. doi:[10.1103/PhysRevLett.123.070503](https://doi.org/10.1103/PhysRevLett.123.070503).
- [63] Xanadu. Pennylane quantum chemistry datasets. <https://pennylane.ai/datasets/>, 2023. Accessed: 2024.
- [64] Julia Kempe, Alexei Kitaev, and Oded Regev. The complexity of the local hamiltonian problem. *SIAM Journal on Computing*, 35(5):1070–1097, 2006. doi:[10.1137/S0097539704445226](https://doi.org/10.1137/S0097539704445226). URL <https://doi.org/10.1137/S0097539704445226>.
- [65] Roberto Oliveira and Barbara M. Terhal. The complexity of quantum spin systems on a two-dimensional square lattice. *Quantum Information & Computation*, 8(10):0900–0924, 2008.
- [66] Sevag Gharibian, Yichen Huang, Zeph Landau, and Seung Woo Shin. Quantum hamiltonian complexity. *Foundations and Trends in Theoretical Computer Science*, 10(3):159–282, 2015. doi:[10.1561/0400000066](https://doi.org/10.1561/0400000066).
- [67] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021. doi:[10.1038/s42254-021-00348-9](https://doi.org/10.1038/s42254-021-00348-9).
- [68] Min-Quan He, Dan-Bo Zhang, and Z. D. Wang. Quantum gaussian filter for exploring ground-state properties. *Physical Review A*, 106(3):032420, 2022. doi:[10.1103/PhysRevA.106.032420](https://doi.org/10.1103/PhysRevA.106.032420).
- [69] JuliaPhysics. Quantum ising model tutorial. [https://juliaphysics.github.io/PhysicsTutorials.jl/tutorials/general/quantum\\_ising/quantum\\_ising.html](https://juliaphysics.github.io/PhysicsTutorials.jl/tutorials/general/quantum_ising/quantum_ising.html). Accessed: 2024.

- [70] Heinz-Peter Breuer and Francesco Petruccione. *The theory of open quantum systems*. Oxford University Press, USA, 2002.
- [71] Vittorio Gorini, Andrzej Kossakowski, and E. C. G. Sudarshan. Completely Positive Dynamical Semigroups of N Level Systems. *J. Math. Phys.*, 17:821, 1976. doi:[10.1063/1.522979](https://doi.org/10.1063/1.522979).
- [72] G. Lindblad. On the generators of quantum dynamical semigroups. *Communications in Mathematical Physics*, 48(2):119–130, June 1976. ISSN 1432-0916. doi:[10.1007/BF01608499](https://doi.org/10.1007/BF01608499).
- [73] Richard Cleve and Chunhao Wang. Efficient quantum algorithms for simulating Lindblad evolution. In *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, volume 80, pages 17:1–17:14. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2017. doi:[10.4230/LIPIcs.ICALP.2017.17](https://doi.org/10.4230/LIPIcs.ICALP.2017.17).
- [74] F. Ciccarello, G. M. Palma, and V. Giovannetti. Collision-model-based approach to non-Markovian quantum dynamics. *Phys. Rev. A*, 87:040103, Apr 2013. doi:[10.1103/PhysRevA.87.040103](https://doi.org/10.1103/PhysRevA.87.040103).
- [75] Matthew Pocanic, Dvira Segal, and Nathan Wiebe. Quantum simulation of Lindbladian dynamics via repeated interactions. *J. Phys. A: Math. Theor.*, 58(30):305302, 2025. doi:[10.1088/1751-8121/adebc4](https://doi.org/10.1088/1751-8121/adebc4).
- [76] Andrew M Childs and Tongyang Li. Efficient simulation of sparse Markovian quantum dynamics. *Quantum Information & Computation*, 17(11-12):901–947, 2017. doi:[10.26421/QIC17.11-12-1](https://doi.org/10.26421/QIC17.11-12-1).
- [77] Xiantao Li and Chunhao Wang. Simulating Markovian open quantum systems using higher-order series expansion. In *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261, pages 87:1–87:20. Schloss Dagstuhl-Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing, 2023. doi:[10.4230/LIPIcs.ICALP.2023.87](https://doi.org/10.4230/LIPIcs.ICALP.2023.87).
- [78] Zhiyan Ding, Xiantao Li, and Lin Lin. Simulating open quantum systems using Hamiltonian simulations. *PRX Quantum*, 5(2):020332, 2024. doi:[10.1103/PRXQuantum.5.020332](https://doi.org/10.1103/PRXQuantum.5.020332).
- [79] Andrew M Childs, Yuan Su, Minh C Tran, Nathan Wiebe, and Shuchen Zhu. Theory of Trotter error with commutator scaling. *Physical Review X*, 11(1):011020, 2021. doi:[10.1103/PhysRevX.11.011020](https://doi.org/10.1103/PhysRevX.11.011020).
- [80] Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. Simulating Hamiltonian dynamics with a truncated Taylor series. *Phys. Rev. Lett.*, 114(9):090502, 2015. doi:[10.1103/PhysRevLett.114.090502](https://doi.org/10.1103/PhysRevLett.114.090502).