ABSTRACT

Title of dissertation: ALGORITHMS FOR QUANTUM SIMULATION: DESIGN, ANALYSIS, IMPLEMENTATION, AND APPLICATION

Yuan Su
Doctor of Philosophy, 2020

Dissertation directed by: Professor Andrew M. Childs
Department of Computer Science

Simulating the Hamiltonian dynamics of quantum systems is one of the most promising applications of digital quantum computers. In this dissertation, we develop an understanding of quantum simulation algorithms concerning their design, analysis, implementation, and application.

We implement three leading simulation algorithms, employing diverse techniques to tighten their error analyses and optimize circuit implementations. We produce concrete resource estimates for simulating a Heisenberg spin system, a problem arising in condensed matter physics that is otherwise difficult to solve on a classical computer. The resulting circuits are orders of magnitude smaller than those for the simplest classically-infeasible instances of factoring and quantum chemistry, suggesting the simulation of spin systems as a promising candidate for an early demonstration of practical quantum computation.

We design new simulation algorithms by using classical randomness. We

show that by simply randomizing how the terms in the Hamiltonian are ordered, one can prove stronger bounds for product formulas and thereby give more efficient quantum simulations. We also develop a classical sampler for time-dependent Hamiltonians, using which we give a simulation algorithm that substantially improves over previous approaches when the Hamiltonian varies significantly with time.

We propose a general theory to analyzing product formulas, an approach to quantum simulation widely used in experimental demonstrations but whose error scaling was poorly understood. Our approach directly exploits the commutativity of Hamiltonian, overcoming the limitations of prior error analyses. We prove new speedups of product formulas for simulating many quantum systems, including simulations of nearest-neighbor lattice systems, second-quantized plane-wave electronic structure, k-local Hamiltonians, rapidly decaying power-law interactions, and clustered Hamiltonians, nearly matching or even outperforming the best previous results in quantum simulation. We accompany our analysis with numerical calculation, which suggests that the bounds also have nearly tight constant prefactors.

We identify applications of quantum simulation to designing other quantum algorithms and improving quantum Monte Carlo methods. We develop an algorithmic framework "quantum singular value transformation" using techniques from quantum simulation and apply it to implement principal component regres-

sion. We also apply our new analysis of product formulas and obtain improved quantum Monte Carlo simulations of the transverse field Ising model and quantum ferromagnets.

ALGORITHMS FOR QUANTUM SIMULATION: DESIGN, ANALYSIS, IMPLEMENTATION, AND APPLICATION

by

Yuan Su

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2020

Advisory Committee:
Professor Andrew M. Childs, Chair/Advisor
Professor Alexey V. Gorshkov
Professor Furong Huang
Professor Norbert M. Linke
Professor Lawrence C. Washington

Dedication

To my parents, for encouraging and supporting me

to pursue my passions.

Acknowledgments

This dissertation would not have been possible without the countless support I received during my study, to which I express my sincere gratitude and appreciation.

I am fortunate to have Andrew Childs as my academic adviser. He introduced me to the research area of quantum simulation—the main theme of this dissertation—when I was wandering at the beginning of my journey. He offered close guidance during my first project but has since pushed me to become a more independent researcher. He encouraged me to not only do research but also do great research, a value that has influenced me throughout my doctoral study. I thank him for letting me interact with other researchers in quantum computing and for providing numerous resources through which this is possible. Apart from occasionally being too nice, Andrew has been an amazing adviser I could constantly learn from and work with.

I am grateful to Dmitri Maslov, John Watrous, Nathan Wiebe, Alexey Gorshkov, Mark Wilde, Dominic Berry, and Ryan Babbush for cosupervising my research projects. In particular, I have learned a lot about the implementation of quantum computation from Dmitri, condensed matter physics from Alexey, quantum Shannon theory from Mark, and quantum simulation from Dominic. I thank Nathan for guiding me through my two quantum simulation projects and for kindly sharing me his top jokes (and bad jokes). I thank Ryan for teaching me quantum

chemistry and for hosting a birthday party during my internship. I feel honored to be able to work with John on our side project, and I am deeply inspired by the level of precision and rigor he chose to pursue in the paper writing.

I cherish the opportunities to work with Yunseong Nam, Neil Julien Ross, András Gilyén, Guang Hao Low, Minh Cong Tran, Xin Wang, and many others not mentioned but still greatly appreciated. Their diverse research backgrounds and areas of expertise have resulted in fruitful discussions and collaborations, from which I have benefited a lot. Their support has made me feel less burdened even in the toughest stage of our projects. I am grateful to them for sharing insights on the challenging problems in my research, and I wish I could be as smart and knowledgeable as my wonderful collaborators.

I have enjoyed the many brainstorms and in-depth discussions with members of the broader quantum information science community, including Fernando G.S.L. Brãndao, Earl Campbell, Richard Cleve, Jens Eisert, Jeongwan Haah, Stephen Jordan, Robin Kothari, Yi-Kai Liu, Jarrod R. McClean, Carl Miller, Māris Ozols, Ben Reichardt, Rolando Somma, Brian Swingle, Ronald de Wolf, Xiaodi Wu, and Anurag Anshu, Chi-Fang Chen, Frédéric Dupuis, Xun Gao, Hsin-Yuan Huang, Mária Kieferová, Natalie Klco, Sam McArdle, Yuval Sanders, Kunal Sharma, Guojing Tian, Leonard Wossnig, Shuchen Zhu. I also had opportunities to visit the quantum research groups at Beijing University of Posts and Telecommunications, Chinese Academy of Sciences, Peking University, Macquarie Univer-

sity, Institute for Quantum Computing, Center for Computation and Technology, Lawrence Berkeley National Laboratory, and Southwest Jiaotong University, and I appreciate their friendliness and hospitality during my visits.

My life as a graduate student would not have been so enjoyable without friends and colleagues at Maryland: Aniruddha Bapat, Dan Carney, Jianxin Chen, Su-Kuan Chu, Abhinav Deshpande, Hong Hao Fu, James Garrison, Andrew Glaudell, Ujjwal Goel, Andrew Guo, Yiming Huang, Shih-Han Hung, Shelby Kimmel, Ali Izadi Rad, Tongyang Li, Cedric Lin, Jin-Peng Liu, Aaron Ostrander, Yuxiang Peng, Eddie Schoute, Troy Sewell, Amr Sharaf, Guowei Sun, Aarthi Sundaram, Chiao-Hsuan Wang, Chutian Wang, Daochen Wang, Xingyao Wu, Pan Xu, Sheng Yang, Penghui Yao, Qi Zhao, Zhenpeng Zhao, and Shaopeng Zhu. During my doctoral study, I had opportunities to take many challenging courses and I thank all the instructors for sharpening my skills and broadening my horizons. I would also like to thank Javiera Caceres and Andrea Svejda for making our research center a nice place for working. Whenever I was looking for assistance, they were always there to offer help.

I had a lot of fun sharing apartment with my roommate Zhouchen Luo over the past five years. Despite him an experimentalist and me a theorist, we never had trouble finding topics to chat about and ways to hang out together. It was also a pleasure to stay with Colin Graber during the Google internship, with whom I had many interesting conversations.

I thank members of my dissertation committee—Andrew Childs, Alexey Gorshkov, Furong Huang, Norbert Linke, and Lawrence Washington—for reading my dissertation and providing useful feedback. I am grateful to Andrew Childs, Howard Elman, and Xiaodi Wu for serving on my preliminary examination committee. I thank Tom Hurst for his help in the preparation of dissertation defense.

Finally, I thank my parents and other relatives for the endless love and support they have given me over the years.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Algorithms for quantum simulation

Simulating the Hamiltonian dynamics of quantum systems is one of the most promising applications of digital quantum computers. The apparent classical intractability of simulating quantum dynamics led Feynman [47] and others to propose the idea of quantum computation. Quantum computers can simulate various physical systems, including condensed matter physics [9], quantum field theory [64], and quantum chemistry [2, 26, 81, 109]. The study of quantum simulation has also led to the discovery of new quantum algorithms, such as algorithms for linear systems [55], differential equations [11], semidefinite optimization [19], formula evaluation [44], quantum walk [30], and ground-state and thermal-state preparation [38, 89], and could ultimately lead to practical applications such as designing new pharmaceuticals, catalysts, and materials [9, 39].

Mathematically, we represent Hamiltonians by Hermitian operators $\mathscr{H}(\tau)$ satisfying $\mathscr{H}(\tau)^\dagger = \mathscr{H}(\tau)$ for $0 \leq \tau \leq t$. The goal of quantum simulation is to use a quantum circuit to approximate the time evolution $\exp_{\mathcal{T}}\left(-i\int_0^t \mathrm{d}\tau\ \mathscr{H}(\tau)\right)$ with error at most ϵ, where $\exp_{\mathcal{T}}$ denotes the time-ordered matrix exponential. The complexity of quantum simulation is then quantified by the number of elementary gates used by the circuit. In the case where the Hamiltonian $\mathscr{H}(\tau) = H$ does not

depend on time, the evolution operator can be represented in closed form as e^{-itH} and quantum simulation can be greatly simplified. This dissertation mainly considers quantum algorithms for simulating time-independent Hamiltonians, although we also discuss the time-dependent case where the problem becomes considerably harder to solve.

In 1996, Lloyd gave the first explicit quantum algorithm for simulating k-local Hamiltonians [70]. His approach is based on product formulas. Specifically, let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be a k-local Hamiltonian (i.e., each H_γ acts nontrivially on $k = \mathcal{O}(1)$ qubits). Assuming H is time-independent, evolution under H for time t is described by the unitary operation $e^{-itH} = e^{-it\sum_{\gamma=1}^{\Gamma} H_\gamma}$. When t is small, this evolution can be well-approximated by the Lie-Trotter formula $\mathscr{S}_1(t) = e^{-itH_\Gamma} \cdots e^{-itH_1}$, where each e^{-itH_γ} can be efficiently implemented on a quantum computer. To simulate for a longer time, we may divide the evolution into r *Trotter steps* and simulate each step with *Trotter error* at most ϵ/r. We choose the *Trotter number* r to be sufficiently large so that the entire simulation achieves an error of at most ϵ. The Lie-Trotter formula only provides a first-order approximation to the evolution, but higher-order approximations are also known from the work of Suzuki and others [18, 100]. A quantum simulation algorithm using product formulas does not require ancilla qubits, making this approach advantageous for near-term experimental demonstration.

Recent studies have considered the broader class of sparse Hamiltonians

[1, 12, 13, 15, 71, 73] and provided alternative simulation algorithms beyond the product-formula approach. Some of these algorithms have nearly linear dependence on the evolution time and logarithmic dependence on the allowed error [13–15, 73, 74, 76], a dramatic improvement over product formulas. In particular, the algorithm based on quantum signal processing [73, 74] is optimal for simulating sparse Hamiltonians with respect to all parameters of interest. The aim of this dissertation is to develop a further understanding of quantum simulation algorithms concerning their design, analysis, implementation, and application. In **Chapter 2**, we give a summary of background material that is necessary for understanding the remaining discussion of this dissertation.

1.2 Circuit implementation

While recent simulation algorithms provide large asymptotic improvement over product formulas, little is known about their practical performance for simulating concrete physical systems. In particular, the constant-factor overhead and extra space requirements may make them uncompetitive with the product-formula approach in practice. This consideration is relevant to near-term quantum simulation experiments, where the number of available qubits and the total number of gates that can be reliably applied in a single run can be significantly limited.

In **Chapter 3**, we address this problem by estimating the resource requirement for simulating a one-dimensional Heisenberg model with a random magnetic

field

$$\sum_{j=1}^{n}(\vec{\Sigma}_j \cdot \vec{\Sigma}_{j+1} + h_j Z_j) \tag{1.1}$$

with periodic boundary conditions (i.e., $\vec{\Sigma}_{n+1} = \vec{\Sigma}_1$), and $h_j \in [-h, h]$ chosen uniformly at random, where $\vec{\Sigma}_j = (X_j, Y_j, Z_j)$ denotes a vector of Pauli X, Y, and Z matrices on qubit j. This model can be simulated to understand condensed matter phenomena, although even a simulation of modest size seems to be infeasible for current classical computers [78]. To produce concrete resource estimate, we consider simulations with $h = 1$, evolution time $t = n$ (the size of the system), and overall accuracy $\epsilon = 10^{-3}$. With all the parameters fixed except n, we compare the complexity of algorithms with respect to the system size.

We implement three leading quantum simulation algorithms: the algorithm using high-order Product Formulas (PF) [12], the algorithm based on truncated Taylor Series (TS) [14], and the recent Quantum-Signal-Processing (QSP) algorithm [73, 74]. We derived concrete error bounds for PF and LCU and resolved an implementation issue of QSP where high precision classical computation is required. We implement these algorithms in a quantum circuit description language called Quipper [51] and process all circuits using an automated tool we developed for large-scale quantum circuit optimization [82]. Although the product-formula algorithm is theoretically surpassed by more sophisticated algorithms, we find that its empirical performance is the best among all the algorithms we study.

The resulting circuits are orders of magnitude smaller than those for the simplest classically-infeasible instances of factoring and quantum chemistry, suggesting the simulation of spin systems as a promising candidate for an early demonstration of practical quantum computation. This chapter is partly based on the following paper:

[34] 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** (2018), no. 38, 9456–9461, arXiv:1711.10980.

1.3 Quantum simulation by randomization

Randomization can be a powerful tool for quantum simulation. For example, Poulin et al. gave improved simulations of time-dependent Hamiltonians by sampling the Hamiltonian at random times [88]. Zhang studied the effect of randomizing the ordering and/or duration of evolutions in a product formula, showing that randomly ordering the summands in the first-order formula in either forward or reverse order can give an improved algorithm [113]. Whether there exist other scenarios in which randomness can improve the performance of quantum simulation remains underexplored.

In **Chapter 4**, we develop a simulation algorithm based on higher-order product formulas, which can achieve significantly better asymptotic performance than the first-order formula. Specifically, we analyze the effect of randomly per-

muting all the Γ summands in the Hamiltonian $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$. The resulting algorithm is not much more complicated than a deterministic product formula, but the savings in the simulation cost are substantial. We further provide numerical evidence suggesting that the randomized approach can be advantageous in practice. This chapter is partly based on the following paper:

[35] Andrew M. Childs, Aaron Ostrander, and Yuan Su, *Faster quantum simulation by randomization*, Quantum **3** (2019), 182, arXiv:1805.08385.

Our focus has so far been on the time-independent Hamiltonian simulation. Simulating a general time-dependent $\mathscr{H}(\tau)$ naturally subsumes the time-independent case, and can be applied to devising quantum control schemes [85], describing quantum chemical reactions [23], and implementing adiabatic quantum algorithms [45]. However, the problem becomes considerably harder and there are fewer quantum algorithms available [13, 14, 65, 76, 88, 110].

In **Chapter 5**, we develop a randomized approach for simulating time-dependent Hamiltonians that is strictly faster than existing algorithms. Specifically, to simulate $\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \alpha_\gamma(\tau) H_\gamma$ where $\|H_\gamma\| \leq 1$, we give a classical sampler whose performance depends on the integration $\int_0^t \mathrm{d}\tau \sum_{\gamma=1}^{\Gamma} |\alpha_\gamma(\tau)|$ rather than the worst-case value $\max_\tau \max_\gamma |\alpha_\gamma(\tau)|$. Our algorithm is thus advantageous when the Hamiltonian varies significantly. We further identify a concrete problem in quantum chemistry—the semi-classical scattering of molecules—for which our algorithm with L^1-norm scaling offers a polynomial speedup over previous

approaches. This chapter is partly based on the following paper:

[16] Dominic W. Berry, Andrew M. Childs, Yuan Su, Xin Wang, and Nathan Wiebe, *Time-dependent Hamiltonian simulation with L^1-norm scaling*, Quantum **4** (2020), 254, arXiv:1906.07115.

1.4 Analysis of product formulas

Product formulas and their generalizations [35, 54, 75, 84] can perform significantly better when the operator summands commute or nearly commute—a unique feature that does not seem to hold for other quantum simulation algorithms [13–15, 25, 73, 74, 76]. This effect has been observed numerically in previous studies of quantum simulations of condensed matter systems [34] and quantum chemistry [8, 90, 108]. An intuitive explanation of this phenomenon comes from truncating the Baker-Campbell-Hausdorff (BCH) series. However, the intuition that the lowest-order terms of the BCH expansion are dominant is surprisingly difficult to justify (and sometimes is not even valid [36, 107]). Thus, previous work established loose Trotter error bounds, leaving a dramatic gap between their provable performance and actual behavior. This gap makes it hard to identify the fastest simulation algorithm and to find optimized implementations for near-term applications of quantum computers.

In **Chapter 6**, we develop a general theory for analyzing product formulas, overcoming the limitations of previous Trotter error analyses. We then identify

a host of its applications to digital quantum simulation in **Chapter 7**, including simulations of nearest-neighbor lattice systems, second-quantized plane-wave electronic structure, k-local Hamiltonians, rapidly decaying power-law interactions, and clustered Hamiltonians, nearly matching or even outperforming the best previous results in quantum simulation.

We further numerically implement our bound for the one-dimensional Heisenberg model (3.1). As aforementioned, this model can be simulated much more efficiently using product formulas, although this efficiency was poorly understood from a theoretical perspective. Here, we give a tight bound that is loose by only a factor of about 5, making significant progress toward a precise characterization of Trotter error. These two chapters are partly based on the following papers:

[36] Andrew M. Childs and Yuan Su, *Nearly optimal lattice simulation by product formulas*, Physical Review Letters **123** (2019), 050503, arXiv:1901.00564.

[37] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu, *A theory of Trotter error*, 2019, arXiv:1912.08854.

1.5 Application of quantum simulation

For a given Hamiltonian H and evolution time t, the goal of quantum simulation is to implement e^{-itH} using a quantum circuit comprised of elementary gates. Restricted to each eigensubspace, the goal is to implement the transformation $\lambda \mapsto e^{-it\lambda}$, which can be done using recent simulation algorithms such as the

Taylor-series algorithm [14] and quantum signal processing [73]. However, quantum simulation is only one of the many examples where quantum computers could offer speedup. Recent studies have considered other problems, such as quantum search and amplitude amplification [52], solving linear equations [55], quantum walks [29] and quantum machine learning [17, 111], where the goal can also be viewed as applying functions to the eigenvalues/singular values of the input matrices encoded in certain form. It is therefore natural to question whether techniques from quantum simulation can be used to design algorithms for other problems.

In **Chapter 8**, we develop an algorithmic framework called "quantum singular value transformation", which applies polynomial functions to the singular values of matrices encoded in a standard form. This framework originates from quantum simulation, but it is applicable to a host of other problems, dramatically simplifying previous analyses and revealing new algorithms that were previously unknown. We prove a spectral theorem for quantum singular value transformation and apply the framework to implement principal component regression in machine learning [48]. This chapter is partly based on the following paper:

[50] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe, *Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics*, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, pp. 193–204, ACM, 2019, arXiv:1806.01838.

Beyond quantum simulation, product formulas can also be applied to quan-

tum Monte Carlo methods, in which the goal is to classically compute certain properties of the Hamiltonian, such as the partition function, the free energy, or the ground energy. Previous work considered quantum Monte Carlo methods for various systems [20, 21], although their analyses do not exploit the commutativity of the Hamiltonian and may thus be improved to give more efficient simulations.

In **Chapter 9**, we apply our analysis of product formulas to improve the performance of quantum Monte Carlo simulation. Our result includes a simulation of the transverse field Ising model, tightening the previous result [20], and a simulation of ferromagnetic quantum spin systems, improving the analysis of [21]. This chapter is partly based on the following paper:

[37] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu, *A theory of Trotter error*, 2019, arXiv:1912.08854.

We conclude the dissertation in Chapter 10 by briefly summarizing our contributions and identifying multiple directions for future work.

Chapter 2: Preliminaries

In this chapter, we summarize useful background material that is necessary for understanding the remaining discussion of the dissertation. Specifically, we introduce notation and terminology in Section 2.1, including various notions of norms and common asymptotic notations. In Section 2.2, we discuss time-ordered evolution operators and their mathematical properties. We then introduce common Hamiltonian input models in Section 2.3. In Section 2.4, we introduce product formulas and establish a simple but loose error bound for the product-formula algorithm. Finally, we review two recent quantum simulation algorithms—the Taylor-series algorithm and the quantum-signal-processing algorithm—in Section 2.5 and Section 2.6.

This chapter is partly based on the following papers:

[34] 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** (2018), no. 38, 9456–9461, arXiv:1711.10980.

[16] Dominic W. Berry, Andrew M. Childs, Yuan Su, Xin Wang, and Nathan Wiebe, *Time-dependent Hamiltonian simulation with L^1-norm scaling*, Quantum **4** (2020), 254, arXiv:1906.07115.

[37] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu, *A theory of Trotter error*, 2019, arXiv:1912.08854.

We assume throughout this dissertation that the reader is familiar with basic notions of quantum computing, including quantum states, quantum operations, and measurements for closed/open quantum systems. Sources to attain this background include the textbook by Nielsen and Chuang [83], as well as the lecture notes by Bacon [10] and Watrous [104]. We also assume a basic familiarity of matrix analysis [60] and linear algebra [6], with the exception of Chapter 8, where familiarity with advanced linear algebra [91] is assumed.

2.1 Notation and terminology

Throughout this dissertation, we consider finite-dimensional complex vector spaces equipped with inner product, where vectors can be represented by their coordinates and operators can be represented by matrices. We use Dirac notation $|\psi\rangle$ to represent unit vectors/pure quantum states and $\langle\phi| = |\phi\rangle^\dagger$ to represent dual vectors, so the scalar $\langle\phi|\psi\rangle$ gives the inner product of $|\psi\rangle$ and $|\phi\rangle$. For a d-dimensional space, we let $\{|j\rangle\}_{j=1}^d$ be an arbitrary but fixed orthonormal basis. For any operator A and orthonormal bases $\mathcal{B}_1$ and $\mathcal{B}_2$, we denote the matrix representation of A as $[A]_{\mathcal{B}_1,\mathcal{B}_2}$. We construct composite spaces by taking tensor product and we drop the symbol $\otimes$ when there is no ambiguity.

Unless otherwise noted, we use lowercase Latin letters to represent scalars, such as the evolution time t, the system size n, and the order of a product formula p. We also use the Greek alphabet to denote scalars, especially when we want

to write a summation like $\sum_{\gamma=1}^{\Gamma}$. We use uppercase Latin letters, such as A, to denote operators. We use scripted uppercase letters, such as $\mathscr{F}(t)$, to denote operator-valued functions.

We organize scalars to form vectors h_γ and tensors $h_{\gamma_1,\ldots,\gamma_k}$. We use standard norms for tensors, including the 1-norm $\|h\|_1 := \sum_{\gamma_1,\ldots,\gamma_k} |h_{\gamma_1,\ldots,\gamma_k}|$, the Euclidean norm (or 2-norm) $\|h\|_2 := \sqrt{\sum_{\gamma_1,\ldots,\gamma_k} |h_{\gamma_1,\ldots,\gamma_k}|^2}$, and the ∞-norm $\|h\|_\infty := \max_{\gamma_1,\ldots,\gamma_k} |h_{\gamma_1,\ldots,\gamma_k}|$. In case there is ambiguity, we use $\vec{h}$ to emphasize the fact that h is a vector (or a tensor more generally).

For an operator A, we use $\|A\|$ to denote its spectral norm—the largest singular value of A. The spectral norm is also known as the operator norm. It is a matrix norm that satisfies the scaling property $\|aA\| = |a| \, \|A\|$, the submultiplicative property $\|AB\| \leq \|A\| \, \|B\|$, and the triangle inequality $\|A + B\| \leq \|A\| + \|B\|$. If A is unitary, then $\|A\| = 1$. We further use $A_{\gamma_1,\ldots,\gamma_k}$ to denote a tensor where each elementary object is an operator. We define a norm of $A_{\gamma_1,\ldots,\gamma_k}$ by taking the spectral norm of each elementary operator and evaluating the corresponding norm of the resulting tensor. For example, we have $\|A\|_1 := \sum_{\gamma_1,\ldots,\gamma_k} \|A_{\gamma_1,\ldots,\gamma_k}\|$ and $\|A\|_\infty := \max_{\gamma_1,\ldots,\gamma_k} \|A_{\gamma_1,\ldots,\gamma_k}\|$.

Let $f, g : \mathbb{R} \to \mathbb{R}$ be functions of real variables. We write $f = O(g)$ if there exist $c, t_0 > 0$ such that $|f(\tau)| \leq c \, |g(\tau)|$ whenever $|\tau| \leq t_0$. Note that we consider the limit when the variable τ approaches zero as opposed to infinity, which is different from the usual setting of algorithmic analysis. For that purpose,

we write $f = \mathcal{O}(g)$ if there exist $c, t_1 > 0$ such that $|f(\tau)| \leq c\,|g(\tau)|$ for all $|\tau| \geq t_1$. When there is no ambiguity, we will use $f = \mathcal{O}(g)$ to also represent the case where $|f(\tau)| \leq c\,|g(\tau)|$ holds for all $\tau \in \mathbb{R}$. We then extend the definition of $\mathcal{O}$ to functions of positive integers and multivariate functions. For example, we use $f(n, t, 1/\epsilon) = \mathcal{O}((nt)^2/\epsilon)$ to mean that $|f(n, t, 1/\epsilon)| \leq c(n\,|t|)^2/\epsilon$ for some $c, n_0, t_0, \epsilon_0 > 0$ and all $|t| \geq t_0$, $0 < \epsilon < \epsilon_0$, and integers $n \geq n_0$. If $\mathscr{F}(\tau)$ is an operator-valued function, we first compute its spectral norm and analyze the asymptotic scaling of $\|\mathscr{F}(\tau)\|$. We write $f = \Omega(g)$ if $g = \mathcal{O}(f)$, and $f = \Theta(g)$ if both $f = \mathcal{O}(g)$ and $f = \Omega(g)$. We use $\widetilde{\mathcal{O}}$ to suppress logarithmic factors in the asymptotic expression and $o(1)$ to represent a positive number that approaches zero as some parameter grows.

Finally, we use $\overleftarrow{\prod}$, $\prod_{\gamma=1}^{\Gamma}$ to denote a product where the elements have increasing indices from right to left and $\overrightarrow{\prod}$, $\prod_{\gamma=\Gamma}^{1}$ vice versa. Under this convention,

$$\prod_{\gamma=1}^{\Gamma} A_\gamma = \overleftarrow{\prod_{\gamma}} A_\gamma = A_\Gamma \cdots A_2 A_1, \qquad \prod_{\gamma=\Gamma}^{1} A_\gamma = \overrightarrow{\prod_{\gamma}} A_\gamma = A_1 A_2 \cdots A_\Gamma. \tag{2.1}$$

We let a summation be zero if its lower limit exceeds its upper limit.

2.2 Time-ordered evolution

Let $\mathscr{H}(\tau)$ be an operator-valued function defined for $0 \leq \tau \leq t$. We say that $\mathscr{U}(\tau)$ is the time-ordered evolution generated by $\mathscr{H}(\tau)$ if $\mathscr{U}(0) = I$ and $\frac{\mathrm{d}}{\mathrm{d}\tau}\mathscr{U}(\tau) = \mathscr{H}(\tau)\mathscr{U}(\tau)$ for $0 \leq \tau \leq t$. In the case where $\mathscr{H}(\tau)$ is anti-Hermitian,

the function $\mathscr{U}(\tau)$ represents the evolution of a quantum system under Hamiltonian $i\mathscr{H}(\tau)$. We not only consider this special case but also study the general case where $\mathscr{H}(\tau)$ can be an arbitrary operator valued function, so that our result can be applied to quantum Monte Carl methods (Chapter 9) as well. Throughout this dissertation, we assume that operator-valued functions are continuous, which guarantees the existence and uniqueness of their generated evolutions [42, p. 12]. We then formally represent the time-ordered evolution $\mathscr{U}(t)$ by $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \mathscr{H}(\tau)\right)$, where $\exp_{\mathcal{T}}$ denotes the time-ordered exponential. In the special case where $\mathscr{H}(\tau) = H$ is constant, the generated evolution is given by an ordinary matrix exponential $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \mathscr{H}(\tau)\right) = e^{tH}$.

In a similar way, we define the time-ordered evolution $\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right)$ generated on an arbitrary interval $t_1 \leq \tau \leq t_2$. Its determinant satisfies [42, p. 9]

$$\det\left(\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right)\right) = e^{\int_{t_1}^{t_2} \mathrm{d}\tau \mathrm{Tr}(\mathscr{H}(\tau))} \neq 0, \tag{2.2}$$

so the inverse $\exp_{\mathcal{T}}^{-1}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right)$ exists; we denote it by $\exp_{\mathcal{T}}\left(\int_{t_2}^{t_1} \mathrm{d}\tau \mathscr{H}(\tau)\right)$. We have thus defined $\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right)$ for every pair of t_1 and t_2 in the domain of $\mathscr{H}(\tau)$.[1] Time-ordered exponentials satisfy the differentiation rule [42, p. 12]

$$\begin{aligned}
\frac{\partial}{\partial t_2} \exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau\ \mathscr{H}(\tau)\right) &= \mathscr{H}(t_2) \exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau\ \mathscr{H}(\tau)\right), \\
\frac{\partial}{\partial t_1} \exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau\ \mathscr{H}(\tau)\right) &= -\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau\ \mathscr{H}(\tau)\right) \mathscr{H}(t_1),
\end{aligned} \tag{2.3}$$

and the multiplicative property [42, p. 11]

$$\exp_{\mathcal{T}}\left(\int_{t_1}^{t_3} \mathrm{d}\tau\ \mathscr{H}(\tau)\right) = \exp_{\mathcal{T}}\left(\int_{t_2}^{t_3} \mathrm{d}\tau\ \mathscr{H}(\tau)\right)\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau\ \mathscr{H}(\tau)\right). \quad (2.4)$$

By definition, the operator-valued function $\mathscr{U}(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \mathscr{H}(\tau)\right)$ satisfies the differential equation $\frac{\mathrm{d}}{\mathrm{d}\tau}\mathscr{U}(\tau) = \mathscr{H}(\tau)\mathscr{U}(\tau)$ with initial condition $\mathscr{U}(0) = I$. We then find the integral equation

$$\mathscr{U}(t) = I + \int_0^t \mathrm{d}\tau\ \mathscr{H}(\tau)\mathscr{U}(\tau) \quad (2.5)$$

by using the fundamental theorem of calculus. We also consider a general differential equation $\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{U}(t) = \mathscr{H}(t)\mathscr{U}(t) + \mathscr{R}(t)$, whose solution is given by the following variation-of-parameters formula:

Lemma 1 (Variation-of-parameters formula [68, Theorem 4.9] [42, p. 17])**.** *Let $\mathscr{H}(\tau)$, $\mathscr{R}(\tau)$ be continuous operator-valued functions defined for $\tau \in \mathbb{R}$. Then the first-order differential equation*

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{U}(t) = \mathscr{H}(t)\mathscr{U}(t) + \mathscr{R}(t), \quad \mathscr{U}(0)\ \textit{known}, \quad (2.6)$$

[1] Alternatively, we may define a time-ordered exponential by its Dyson series or by a convergent sequence of products of ordinary matrix exponentials, and verify that this alternative definition satisfies a certain differential equation. We prefer the differential-equation definition since it is more versatile for the analysis in this dissertation.

has a unique solution given by the variation-of-parameters formula

$$\mathscr{U}(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{H}(\tau)\right)\mathscr{U}(0) + \int_0^t \mathrm{d}\tau_1\ \exp_{\mathcal{T}}\left(\int_{\tau_1}^t \mathrm{d}\tau_2\ \mathscr{H}(\tau_2)\right)\mathscr{R}(\tau_1). \quad (2.7)$$

Let $\mathscr{H}(\tau) = \mathscr{A}(\tau) + \mathscr{B}(\tau)$ be a continuous operator-valued function with two summands defined for $0 \leq \tau \leq t$. Then, the evolution under $\mathscr{H}(\tau)$ can be seen as evolution under the rotated operator $\exp_{\mathcal{T}}^{-1}\left(\int_0^\tau \mathrm{d}\tau_2 \mathscr{A}(\tau_2)\right)\mathscr{B}(\tau)\exp_{\mathcal{T}}\left(\int_0^\tau \mathrm{d}\tau_2 \mathscr{A}(\tau_2)\right)$, followed by another evolution under $\mathscr{A}(\tau)$ that rotates back to the original frame [76]. This is known as the "interaction-picture" representation in quantum mechanics and is formally stated in the following lemma.

Lemma 2 (Time-ordered evolution in the interaction picture)**.** *Let $\mathscr{H}(\tau) = \mathscr{A}(\tau) + \mathscr{B}(\tau)$ be an operator-valued function defined for $\tau \in \mathbb{R}$ with continuous summands $\mathscr{A}(\tau)$ and $\mathscr{B}(\tau)$. Then*

$$\begin{aligned}\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{H}(\tau)\right) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{A}(\tau)\right) \\ \cdot \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau_1\ \exp_{\mathcal{T}}^{-1}\left(\int_0^{\tau_1} \mathrm{d}\tau_2\ \mathscr{A}(\tau_2)\right)\right. \\ \left. \cdot\, \mathscr{B}(\tau_1)\exp_{\mathcal{T}}\left(\int_0^{\tau_1} \mathrm{d}\tau_2\ \mathscr{A}(\tau_2)\right)\right).\end{aligned} \quad (2.8)$$

Proof. A simple calculation shows that the right-hand side of the above equation satisfies the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{U}(t) = \mathscr{H}(t)\mathscr{U}(t) \quad (2.9)$$

with initial condition $\mathscr{U}(0) = I$. The lemma then follows as $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \mathscr{H}(\tau)\right)$ is the unique solution to this differential equation. □

For any continuous $\mathscr{H}(\tau)$, the evolution $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \mathscr{H}(\tau)\right)$ it generates is invertible and continuously differentiable. Conversely, the following lemma asserts that any operator-valued function that is invertible and continuously differentiable is a time-ordered evolution generated by some continuous function.

Lemma 3 (Fundamental theorem of time-ordered evolution [42, p. 20])**.** *The following statements regarding an operator-valued function $\mathscr{U}(\tau)$ ($\tau \in \mathbb{R}$) are equivalent:*

1. *$\mathscr{U}(\tau)$ is invertible and continuously differentiable;*
2. *$\mathscr{U}(\tau) = \exp_{\mathcal{T}}\left(\int_0^\tau \mathrm{d}\tau_1 \mathscr{H}(\tau_1)\right)\mathscr{U}(0)$ for some continuous operator-valued function $\mathscr{H}(\tau)$.*

Furthermore, in the second statement, $\mathscr{H}(\tau) = \left(\frac{\mathrm{d}}{\mathrm{d}\tau}\mathscr{U}(\tau)\right)\mathscr{U}^{-1}(\tau)$ is uniquely determined.

Finally, we bound the spectral norm of a time-ordered evolution $\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right)$ and the distance between two evolutions.

Lemma 4 (Spectral-norm bound for time-ordered evolution [42, p. 28])**.** *Let $\mathscr{H}(\tau)$ be a continuous operator-valued function defined on $\mathbb{R}$. Then,*

1. $\left\|\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right)\right\| \leq e^{\left|\int_{t_1}^{t_2} \mathrm{d}\tau \|\mathscr{H}(\tau)\|\right|}$; *and*

2. $\left\|\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right)\right\| = 1$ *if* $\mathscr{H}(\tau)$ *is anti-Hermitian.*

Corollary 5 (Distance bound for time-ordered evolutions [102, Appendix B])**.** *Let* $\mathscr{H}(\tau)$ *and* $\mathscr{G}(\tau)$ *be continuous operator-valued functions defined on* $\mathbb{R}$*. Then,*

1. $\left\|\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right) - \exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{G}(\tau)\right)\right\| \leq \left|\int_{t_1}^{t_2} \mathrm{d}\tau \left\|\mathscr{H}(\tau) - \mathscr{G}(\tau)\right\|\right| \cdot e^{\left|\int_{t_1}^{t_2} \mathrm{d}\tau (\|\mathscr{H}(\tau)\| + \|\mathscr{G}(\tau)\|)\right|}$*; and*

2. $\left\|\exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{H}(\tau)\right) - \exp_{\mathcal{T}}\left(\int_{t_1}^{t_2} \mathrm{d}\tau \mathscr{G}(\tau)\right)\right\| \leq \left|\int_{t_1}^{t_2} \mathrm{d}\tau \left\|\mathscr{H}(\tau) - \mathscr{G}(\tau)\right\|\right|$ *if* $\mathscr{H}(\tau)$ *and* $\mathscr{G}(\tau)$ *are anti-Hermitian.*

2.3 Hamiltonian input models

Quantum simulation algorithms may have different performance depending on the choice of the input model of Hamiltonians. In this section, we describe several input models that are commonly used in previous work. We consider the general case where Hamiltonians are time-dependent; the time-independent case can be handled similarly by dropping the time dependence.

Let $\mathscr{H}(\tau)$ be a time-dependent Hamiltonian defined for $0 \leq \tau \leq t$. In the linear-combination (LC) model, we assume that the Hamiltonian admits the decomposition

$$\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \mathscr{H}_\gamma(\tau), \tag{2.10}$$

where the Hermitian-valued functions $\mathscr{H}_\gamma(\tau)$ are continuous and can be efficiently exponentiated on a quantum computer. Such a setting is common in the simulation

of condensed matter physics and quantum chemistry. We will use this model when we develop circuit implementation of product formulas in Chapter 3 and design randomized quantum simulation algorithms in Chapter 4 and Chapter 5.

A variant of the LC model is the linear-combination-of-unitaries (LCU) model. In this case, the Hamiltonian $\mathscr{H}(\tau)$ has the form

$$\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \alpha_\gamma(\tau) H_\gamma, \tag{2.11}$$

where the coefficients $\alpha_\gamma(\tau) \geq 0$ are continuously differentiable and the matrices H_γ are both unitary and Hermitian. We assume that the coefficients $\alpha_\gamma(\tau)$ can be efficiently computed by a classical oracle, and we ignore the classical cost of implementing such an oracle. We further assume that each $|0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes H_\gamma$ can be efficiently implemented. We will use this model when we discuss the Taylor-series algorithm and the quantum-signal-processing algorithm in Chapter 3. In both the LC and the LCU model, we quantify the complexity of a simulation algorithm by the number of elementary gates it uses.

Another common input model is the sparse-matrix (SM) model. We say that $\mathscr{H}(\tau)$ is d-sparse if the number of nonzero matrix elements within each row and column throughout the entire interval $[0, t]$ is at most d. We assume that the locations of the nonzero matrix elements are time independent. Access to the

Hamiltonian is given through the oracles

$$\begin{aligned} |j, s\rangle &\mapsto |j, \mathrm{col}(j, s)\rangle, \\ |\tau, j, k, z\rangle &\mapsto |\tau, j, k, z \oplus \mathscr{H}_{jk}(\tau)\rangle. \end{aligned} \tag{2.12}$$

Here, $\mathrm{col}(j, s)$ returns the column index of the sth element in the jth row that may be nonzero over the entire time interval $[0, t]$. We quantify the complexity of a quantum simulation algorithm by the number of oracular queries it makes, together with the number of additional elementary gates it requires.

Although we will not consider the SM model in the remaining part of the dissertation, much of our result can be translated to that model. In fact, as the following lemma shows, a d-sparse time-independent Hamiltonian can be efficiently decomposed as a sum of 1-sparse terms.

Lemma 6 (Decomposition of sparse Hamiltonians [13, Lemma 4.3 and 4.4])**.** *Let H be a time-independent d-sparse Hamiltonian accessed through oracles. Define $\|A\|_{\max}$ as the largest matrix element of A in absolute value. Then*

1. *there exists a decomposition $H = \sum_{j=1}^{d^2} H_j$, where each H_j is 1-sparse with $\|H_j\|_{\max} \leq \|H\|_{\max}$, and a query to any H_j can be simulated with $\mathcal{O}(1)$ queries to H; and*

2. *for any $\gamma > 0$, there exists an approximate decomposition[2] $\left\|H - \gamma \sum_{j=1}^{\eta} G_j\right\|_{\max}$*

[2]Reference [13] uses [13, Lemma 4.3] and the triangle inequality to show that $\left\|H - \gamma \sum_{j=1}^{\eta} G_j\right\|_{\max} \leq \sqrt{2}\gamma d^2$. However, this bound can be tightened to $\sqrt{2}\gamma$, since the max-norm distance depends on the largest error from rounding off the d^2 1-sparse matrices.

$\leq \sqrt{2}\gamma$, *where* $\eta = \mathcal{O}\big(d^2 \|H\|_{\max}/\gamma\big)$, *each* G_j *is* 1*-sparse with eigenvalues* ± 1, *and a query to any* G_j *can be simulated with* $\mathcal{O}(1)$ *queries to* H.

2.4 Product formulas

Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be a time-independent operator consisting of Γ summands, so that the evolution generated by H is $e^{t\sum_{\gamma=1}^{\Gamma} H_\gamma}$. Product formulas provide a convenient way of decomposing such an evolution into a product of exponentials of individual H_γ. Examples of product formulas include the first-order Lie-Trotter formula

$$\mathscr{S}_1(t) := e^{tH_\Gamma} \cdots e^{tH_1} \tag{2.13}$$

and higher-order Suzuki formulas [100] defined recursively via

$$\begin{aligned} \mathscr{S}_2(t) &:= e^{\frac{t}{2}H_1} \cdots e^{\frac{t}{2}H_\Gamma} e^{\frac{t}{2}H_\Gamma} \cdots e^{\frac{t}{2}H_1}, \\ \mathscr{S}_{2k}(t) &:= \mathscr{S}_{2k-2}(u_k t)^2 \, \mathscr{S}_{2k-2}((1-4u_k)t) \, \mathscr{S}_{2k-2}(u_k t)^2, \end{aligned} \tag{2.14}$$

where $u_k := 1/(4 - 4^{1/(2k-1)})$.

In general, we can write a product formula as

$$\mathscr{S}(t) := \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}, \tag{2.15}$$

where the coefficients $a_{(\upsilon,\gamma)}$ are real numbers. The parameter Υ denotes the number of *stages* of the formula; for the Suzuki formula $\mathscr{S}_{2k}(t)$, we have $\Upsilon = 2 \cdot 5^{k-1}$.

The permutation π_υ controls the ordering of operator summands within stage υ of the formula. For Suzuki's constructions, we alternately reverse the ordering of summands between neighboring stages, but other formulas may use general permutations. For simplicity, we will fix Υ, π_υ and assume that the coefficients $a_{(\upsilon,\gamma)}$ are uniformly bounded by 1 in absolute value. We then consider the performance of the product formula with respect to the input operator summands H_γ (for $\gamma = 1, \ldots, \Gamma$) and the evolution time t.

Product formulas provide a good approximation to the ideal evolution when the time t is small. Specifically, a pth-order formula $\mathscr{S}(t)$ satisfies

$$\mathscr{S}(t) = e^{tH} + O\left(t^{p+1}\right). \tag{2.16}$$

This asymptotic analysis gives the correct error scaling with respect to t, but the dependence on the H_γ is ignored, so it does not provide a full characterization of Trotter error. This issue was addressed in the work of Berry, Ahokas, Cleve, and Sanders [12], who gave a concrete error bound for product formulas with dependence on both t and H_γ. Their original bound depends on the ∞-norm $\Gamma \max_\gamma \|H_\gamma\|$, although it is not hard to improve this to the 1-norm scaling $\sum_{\gamma=1}^{\Gamma} \|H_\gamma\|$. We prove a new error bound in the lemma below; for real-time evolutions, this improves a multiplicative factor of $e^{t\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|}$ over the best previous analysis [75, Eq. (13)].

Lemma 7 (Trotter error with 1-norm scaling)**.** *Let* $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ *be an operator consisting of* Γ *summands and* $t \geq 0$*. Let* $\mathscr{S}(t) = \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}$ *be a pth-order product formula. Then,*

$$\left\| \mathscr{S}(t) - e^{tH} \right\| = \mathcal{O}\left(\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| \, t \right)^{p+1} e^{t\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|} \right). \tag{2.17}$$

Furthermore, if H_γ *are anti-Hermitian,*

$$\left\| \mathscr{S}(t) - e^{tH} \right\| = \mathcal{O}\left(\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| \, t \right)^{p+1} \right). \tag{2.18}$$

Proof. Since $\mathscr{S}(t)$ is a pth-order formula, we know from [36, Supplementary Lemma 1] that the error $\mathscr{A}(t) = \mathscr{S}(t) - e^{tH}$ satisfies $\mathscr{A}(0) = \mathscr{A}'(0) = \cdots = \mathscr{A}^{(p)}(0) = 0$. By Taylor's theorem,

$$\mathscr{S}(t) - e^{tH} = (p+1) \int_0^1 \mathrm{d}u \, (1-u)^p \frac{t^{p+1}}{(p+1)!} \big(\mathscr{S}^{(p+1)}(ut) - H^{p+1} e^{utH} \big), \tag{2.19}$$

where

$$\begin{aligned} &\mathscr{S}^{(p+1)}(ut) \\ &= \sum_{q_{(1,1)}+\cdots+q_{(\Upsilon,\Gamma)}=p+1} \binom{p+1}{q_{(1,1)} \ \cdots \ q_{(\Upsilon,\Gamma)}} \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} \left(a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)} \right)^{q_{(\upsilon,\gamma)}} e^{uta_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}. \end{aligned} \tag{2.20}$$

The spectral norms of $\mathscr{S}^{(p+1)}(ut)$ and $H^{p+1}e^{utH}$ can be bounded as

$$\begin{aligned}
\left\|\mathscr{S}^{(p+1)}(ut)\right\| &\leq \sum_{q_{(1,1)}+\cdots+q_{(\Upsilon,\Gamma)}=p+1} \binom{p+1}{q_{(1,1)} \ \cdots \ q_{(\Upsilon,\Gamma)}} \prod_{\upsilon=1}^{\Upsilon}\prod_{\gamma=1}^{\Gamma} \left\|H_{\pi_\upsilon(\gamma)}\right\|^{q_{(\upsilon,\gamma)}} e^{t\left\|H_{\pi_\upsilon(\gamma)}\right\|} \\
&= \left(\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|\right)^{p+1} e^{t\Upsilon\sum_{\gamma=1}^{\Gamma}\|H_\gamma\|}, \\
\left\|H^{p+1}e^{utH}\right\| &\leq \left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\|\right)^{p+1} e^{t\sum_{\gamma=1}^{\Gamma}\|H_\gamma\|}.
\end{aligned} \tag{2.21}$$

Applying these bounds to the Taylor expansion, we find that

$$\begin{aligned}
&\left\|\mathscr{S}(t) - e^{tH}\right\| \\
\leq &\frac{t^{p+1}}{(p+1)!}\left[\left(\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|\right)^{p+1} e^{t\Upsilon\sum_{\gamma=1}^{\Gamma}\|H_\gamma\|} + \left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\|\right)^{p+1} e^{t\sum_{\gamma=1}^{\Gamma}\|H_\gamma\|}\right] \\
= &\mathcal{O}\left(\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| t\right)^{p+1} e^{t\Upsilon\sum_{\gamma=1}^{\Gamma}\|H_\gamma\|}\right).
\end{aligned} \tag{2.22}$$

The special case where H_γ are anti-Hermitian can be proved in a similar way, except we directly evaluate the spectral norm of a matrix exponential to 1. □

The above bound on the Trotter error works well for small t. To simulate anti-Hermitian H_γ for a large time, we divide the evolution into r steps and apply the product formula within each step. The overall simulation has error

$$\left\|\mathscr{S}^r(t/r) - e^{tH}\right\| = \mathcal{O}\left(\frac{\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| t\right)^{p+1}}{r^p}\right). \tag{2.23}$$

To simulate with accuracy ϵ, it suffices to choose

$$r = \mathcal{O}\left(\frac{\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| t\right)^{1+1/p}}{\epsilon^{1/p}}\right). \tag{2.24}$$

We have thus proved:

Corollary 8 (Trotter number with 1-norm scaling)**.** *Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be an operator consisting of Γ summands with H_γ anti-Hermitian and $t \geq 0$. Let $\mathscr{S}(t)$ be a pth-order product formula. Then, we have $\left\|\mathscr{S}^r(t/r) - e^{tH}\right\| = \mathcal{O}(\epsilon)$ provided*

$$r = \mathcal{O}\left(\frac{\left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| t\right)^{1+1/p}}{\epsilon^{1/p}}\right). \tag{2.25}$$

Note that the above analysis only uses information about the norms of the summands. In the extreme case where all H_γ commute, the Trotter error becomes zero but the above bound can be arbitrarily large. We address this issue by developing a new analysis in Chapter 6 that leverages information about commutation of the H_γs. We then analyze the performance of product formulas for simulating various physical systems in Chapter 7 and discuss applications to quantum Monte Carlo methods in Chapter 9.

2.5 Taylor-series algorithm

Let $H = \sum_{\gamma=1}^{\Gamma} \alpha_\gamma H_\gamma$ be a Hamiltonian in the LCU model such that H_γ are both unitary and Hermitian and $\alpha_\gamma > 0$. We assume that each controlled operation

$|0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes H_\gamma$ can be implemented with cost $\mathcal{O}(1)$. The Taylor-series algorithm of [14] directly implements the (truncated) Taylor series of the evolution operator e^{-itH} for a carefully-chosen constant time, and repeats that procedure until the entire evolution time has been simulated.

Denote the Taylor series for the evolution up to time t, truncated at order K, by

$$\tilde{\mathscr{U}}(t) := \sum_{\kappa=0}^{\mathrm{K}} \frac{(-itH)^\kappa}{\kappa!}. \tag{2.26}$$

For sufficiently large K, the operator $\tilde{\mathscr{U}}(t)$ is a good approximation of e^{-itH}. By the definition $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$, we can rewrite $\tilde{\mathscr{U}}(t)$ as a linear combination of unitaries, namely

$$\begin{aligned}
\tilde{\mathscr{U}}(t) &= \sum_{\kappa=0}^{\mathrm{K}} \frac{(-itH)^\kappa}{\kappa!} && (2.27)\\
&= \sum_{\kappa=0}^{\mathrm{K}} \sum_{\gamma_1,\ldots,\gamma_\kappa=1}^{\Gamma} \frac{t^\kappa}{\kappa!} \alpha_{\gamma_1} \cdots \alpha_{\gamma_\kappa} (-i)^\kappa H_{\gamma_1} \cdots H_{\gamma_\kappa} && (2.28)\\
&= \sum_{\xi=0}^{\Xi-1} \beta_\xi \tilde{V}_\xi, && (2.29)
\end{aligned}$$

for $\Xi = \sum_{\kappa=0}^{\mathrm{K}} \Gamma^\kappa$, where the $\tilde{V}_\xi$ are products of the form $(-i)^\kappa H_{\gamma_1} \cdots H_{\gamma_\kappa}$, and the β_ξ are the corresponding coefficients such that $\beta_\xi > 0$. (For notational convenience, we omit the dependence of β_ξ on t.) The Taylor-series algorithm effectively implements this linear combination on a quantum computer.

To do this, we introduce

$$B|0\rangle := \frac{1}{\sqrt{s}} \sum_{\xi=0}^{\Xi-1} \sqrt{\beta_\xi}|\xi\rangle, \qquad \text{select}(\tilde{V}) := \sum_{\xi=0}^{\Xi-1} |\xi\rangle\langle\xi| \otimes \tilde{V}_\xi \tag{2.30}$$

and define

$$W := (B^\dagger \otimes I)\,\text{select}(\tilde{V})(B \otimes I), \tag{2.31}$$

where

$$s := \sum_{\xi=0}^{\Xi-1} \beta_\xi = \sum_{\kappa=0}^{\mathrm{K}} \frac{(\|\alpha\|_1 t)^\kappa}{\kappa!} \tag{2.32}$$

and $\|\alpha\|_1 := \alpha_1 + \cdots + \alpha_\Gamma$. It is easy to see that $(\langle 0| \otimes I)W(|0\rangle \otimes I) \propto \tilde{\mathscr{U}}(t)$. More precisely, we have

$$W|0\rangle|\psi\rangle = \frac{1}{s}|0\rangle\tilde{\mathscr{U}}(t)|\psi\rangle + \sqrt{1 - \frac{1}{s^2}}|\Phi\rangle \tag{2.33}$$

for some $|\Phi\rangle$ whose ancillary state is supported in the subspace orthogonal to $|0\rangle$. To boost the amplitude to perform the desired operation, we use the isometry

$$-WRW^\dagger RW(|0\rangle \otimes I) \tag{2.34}$$

where $R := (I - 2|0\rangle\langle 0|) \otimes I$.

To implement evolution according to H nearly deterministically, we consider evolution for time $t_{\text{seg}} := \ln 2 / \|\alpha\|_1$. The overall evolution is realized as a sequence of $r := \lceil t/t_{\text{seg}} \rceil$ segments, where the first $r-1$ segments each evolve the state for time t_{seg} and the final segment evolves the state for time $t_{\text{rem}} := t - (r-1)t_{\text{seg}}$. It can be shown that there is a choice of K with

$$\mathrm{K} = \mathcal{O}\left(\frac{\log(\|\alpha\|_1 t_{\text{seg}}/\epsilon)}{\log\log(\|\alpha\|_1 t_{\text{seg}}/\epsilon)}\right) \tag{2.35}$$

such that

$$\left\| -(\langle 0| \otimes I) W R W^\dagger R W (|0\rangle \otimes I) - \exp(-i t_{\text{seg}} H) \right\| = \mathcal{O}\left(\epsilon/r\right). \tag{2.36}$$

The evolution for the remaining time t_{rem} can be performed by rotating an ancilla qubit to artificially increase the duration of the segment. Specifically, provided $s < 2$, we can introduce an ancilla register in the state $|0\rangle$ and apply the rotation

$$|0\rangle \mapsto \frac{s}{2}|0\rangle + \sqrt{1 - \frac{s^2}{4}}|1\rangle. \tag{2.37}$$

Together with the the unitary operator W, this implements the transformation

$$|0\rangle|\psi\rangle \mapsto \frac{1}{2}|00\rangle \tilde{\mathscr{U}}(t)|\psi\rangle + \frac{\sqrt{3}}{2}|\Phi'\rangle \tag{2.38}$$

for some normalized state $|\Phi'\rangle$ with $(\langle 00| \otimes I)|\Phi'\rangle = 0$. Then we can proceed as

before, but with $s = 2$. Indeed, we also perform a similar rotation for the initial $r - 1$ segments to ensure that they have $s = 2$ instead of a slightly smaller value. The asymptotic gate complexity of this simulation algorithm is [14]

$$\mathcal{O}\left(\Gamma \log \Gamma \left\|\alpha\right\|_1 t \frac{\log(\left\|\alpha\right\|_1 t/\epsilon)}{\log\log(\left\|\alpha\right\|_1 t/\epsilon)}\right). \tag{2.39}$$

2.6 Quantum-signal-processing algorithm

Now we summarize the quantum-signal-processing algorithm of Low and Chuang [73, 74]. Again we consider a Hamiltonian in the LCU model $H = \sum_{\gamma=1}^{\Gamma} \alpha_\gamma H_\gamma$, where H_γ are both unitary and Hermitian and $\alpha_\gamma > 0$. We have

$$\frac{H}{\left\|\alpha\right\|_1} = (\langle G| \otimes I)\,\mathrm{select}(H)(|G\rangle \otimes I), \tag{2.40}$$

where

$$\mathrm{select}(H) := \sum_{\gamma=1}^{\Gamma} |\gamma\rangle\langle\gamma| \otimes H_\gamma, \qquad |G\rangle := \frac{1}{\sqrt{\left\|\alpha\right\|_1}} \sum_{\gamma=1}^{\Gamma} \sqrt{\alpha_\gamma}|\gamma\rangle. \tag{2.41}$$

Low and Chuang's concept of *qubitization* [74] relates the spectral decompositions of $H/\left\|\alpha\right\|_1$ and

$$-iQ := -i\big((2|G\rangle\langle G| - I) \otimes I\big)\,\mathrm{select}(H). \tag{2.42}$$

Specifically, let $H/\left\|\alpha\right\|_1 = \sum_\lambda \lambda|\lambda\rangle\langle\lambda|$ be a spectral decomposition of H/α, where the sum runs over all eigenvalues of $H/\left\|\alpha\right\|_1$. By the triangle inequality, $\|H\| \leq \left\|\alpha\right\|_1$, i.e., $|\lambda| \leq 1$. For each eigenvalue $\lambda \in (-1, 1)$ of $H/\left\|\alpha\right\|_1$, the qubitization theorem [74, Theorem 2] asserts that $-iQ$ has two corresponding eigenvalues

$$\lambda_\pm = \mp\sqrt{1-\lambda^2} - i\lambda = \mp e^{\pm i \arcsin \lambda} \tag{2.43}$$

with eigenvectors $|\lambda_\pm\rangle = (|G_\lambda\rangle \pm i|G_\lambda^\perp\rangle)/\sqrt{2}$, where

$$|G_\lambda\rangle := |G\rangle \otimes |\lambda\rangle, \qquad |G_\lambda^\perp\rangle := \frac{\lambda|G_\lambda\rangle - \mathrm{select}(H)|G_\lambda\rangle}{\sqrt{1-\lambda^2}}. \tag{2.44}$$

(Eigenvalues $\lambda_\pm = \pm 1$ correspond to degenerate cases that can be analyzed separately.)

The signal-processing algorithm applies a sequence of operations called *phased iterates*. We introduce an additional ancilla qubit and define the operator

$$V_\phi := (e^{-i\phi\sigma^z/2} \otimes I)\big(|+\rangle\langle+| \otimes I + |-\rangle\langle-| \otimes (-iQ)\big)(e^{i\phi\sigma^z/2} \otimes I) \tag{2.45}$$

for any $\phi \in \mathbb{R}$. Let $-iQ = \sum_\nu e^{i\theta_\nu}|\nu\rangle\langle\nu|$ be a spectral decomposition of $-iQ$, where the sum runs over ν labeling all eigenvectors of $-iQ$. As described above, each eigenvalue $\lambda \in (-1, 1)$ of $H/\left\|\alpha\right\|_1$ corresponds to two eigenvalues $e^{i\theta_{\lambda_\pm}}$ of $-iQ$, where $\theta_{\lambda_+} = \arcsin(\lambda) + \pi$ and $\theta_{\lambda_-} = -\arcsin(\lambda)$. Eigenvalues ± 1 of

$-iQ$ correspond to degenerate cases that can be handled separately. The remaining eigenspaces cannot be reached during any execution of the quantum-signal-processing algorithm, so we can neglect them. Then one can show that

$$V_\phi = \sum_\nu e^{i\theta_\nu/2} R_\phi(\theta_\nu) \otimes |\nu\rangle\langle\nu| \tag{2.46}$$

where

$$R_\phi(\theta) := e^{-i\theta\Sigma_\phi/2}, \qquad \Sigma_\phi := \cos(\phi)X + \sin(\phi)Y. \tag{2.47}$$

Thus each eigenvalue $e^{i\theta_\nu}$ of $-iQ$ is manifested in V_ϕ as an SU(2) operator $R_\phi(\theta_\nu)$ acting on the ancilla qubit.

For any positive even integer m, composing gates with the same rotation amplitude θ but with varying phases $\phi_1, \ldots, \phi_m$ yields

$$R_{\phi_m}(\theta) \cdots R_{\phi_1}(\theta) = a(\cos\tfrac{\theta}{2})\, I + ib(\cos\tfrac{\theta}{2})\, Z + i\cos\tfrac{\theta}{2} c(\sin\tfrac{\theta}{2})\, X + i\cos\tfrac{\theta}{2} d(\sin\tfrac{\theta}{2})\, Y \tag{2.48}$$

for polynomials a, b, c, d of degree at most m. For quantum simulation, only the polynomials a and c are used. This component can be extracted by preparing the ancilla qubit in the state $|+\rangle$, composing the primitive rotations, and postselecting the ancilla qubit in the state $|+\rangle$. The unwanted factor $e^{i\theta_\nu/2}$ may be canceled by

alternating between V_ϕ and $V_{\phi+\pi}^\dagger$, giving

$$V_{\vec{\phi}} := V_{\phi_m+\pi}^\dagger V_{\phi_{M-1}} \cdots V_{\phi_2+\pi}^\dagger V_{\phi_1}. \tag{2.49}$$

To perform Hamiltonian simulation, we implement a function of θ that converts the eigenvalue $e^{i\theta_{\lambda_\pm}}$ of $-iQ$ to the desired phase $e^{-i\lambda t}$, namely the Jacobi-Anger expansion

$$e^{i\sin(\theta)t} = \sum_{k=-\infty}^{\infty} j_k(t) e^{ik\theta}. \tag{2.50}$$

To do this with a polynomial of degree m, we truncate the expansion at order $q := \frac{m}{2} + 1$, giving an approximation with error at most [15]

$$2\sum_{k=q}^{\infty} |j_k(t)| \leq \frac{4t^q}{2^q q!}. \tag{2.51}$$

The angles $\phi_1, \ldots, \phi_m$ that realize this expansion can be computed by an efficient classical procedure (see Lemmas 1 and 3 of [77]).

To simulate evolution of an initial state $|\psi\rangle$, we apply V to the state $|+\rangle \otimes |G\rangle \otimes |\psi\rangle$ and postselects the ancilla register of the output on the state $|+\rangle \otimes |G\rangle$. This procedure simulates the desired evolution with error at most

$$8\frac{4(\|\alpha\|_1 t)^q}{2^q\, q!} \leq \epsilon. \tag{2.52}$$

To achieve simulation for time t and error ϵ, the quantum-signal-processing algo-

rithm uses

$$m = \mathcal{O}\left(\|\alpha\|_1 \, t + \frac{\log(1/\epsilon)}{\log\log(1/\epsilon)} \right) \tag{2.53}$$

phased iterates [73]. For each phased iterate, the dominant part is the select(H) subroutine, which is straightforward to implement with $\mathcal{O}(\Gamma \log \Gamma)$ elementary gates. Overall, we see that the asymptotic gate count is

$$\mathcal{O}\left(\Gamma \log \Gamma \left(\|\alpha\|_1 \, t + \frac{\log(1/\epsilon)}{\log\log(1/\epsilon)} \right) \right). \tag{2.54}$$

Chapter 3: Circuit implementation

In this chapter, we implement three leading simulation algorithms: the product-formula algorithm, the Taylor-series algorithm, and the quantum-signal-processing algorithm. We introduce the target system—a one-dimensional nearest-neighbor Heisenberg model with a random magnetic field in the z direction—in Section 3.1 and describe input models through which quantum algorithms access the target Hamiltonian. We employing diverse techniques to develop concrete error analyses and optimize circuit implementations in Section 3.2, Section 3.3, and Section 3.4. We discuss the results in Section 3.5.

This chapter is partly based on the following paper:

[34] 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** (2018), no. 38, 9456–9461, arXiv:1711.10980.

3.1 Target system

We consider a one-dimensional nearest-neighbor Heisenberg model with a random magnetic field in the z direction. This model is described by the Hamil-

tonian

$$\sum_{j=1}^{n}(\vec{\Sigma}_j \cdot \vec{\Sigma}_{j+1} + h_j Z_j) \tag{3.1}$$

where $\vec{\Sigma}_j = (X_j, Y_j, Z_j)$ denotes a vector of Pauli X, Y, and Z matrices on qubit j. We impose periodic boundary conditions (i.e., $\vec{\Sigma}_{n+1} = \vec{\Sigma}_1$), and $h_j \in [-h, h]$ is chosen uniformly at random. This Hamiltonian has been considered in recent studies of self-thermalization and many-body localization. Despite intensive numerical investigation, the details of a transition between thermal and localized phases remain poorly understood. Indeed, the most extensive numerical study we are aware of was restricted to at most 22 spins [78].

We focus on the cost of simulating the dynamics of Heisenberg model on a quantum computer, as this is the dominant cost in several quantum simulation proposals for exploring self-thermalization [93, 94, 97]. To produce concrete resource estimate, we consider simulations with $h = 1$, evolution time $t = n$ (the number of spins in the system), and overall accuracy $\epsilon = 10^{-3}$, and express the complexity of quantum simulation as a function of n.

We now discuss input models for the Heisenberg Hamiltonian. Note that the Hamiltonian in (3.1) is already expressed as a linear combination of operators, each of which is both unitary and Hermitian. Therefore, the input Hamiltonian can be analyzed in both the LC model and the LCU model introduced in Section 2.3. In particular, each summand of (3.1) has the form $U_j U_{j+1}$ and the controlled operation

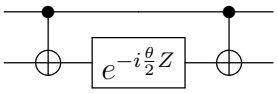

Figure 3.1: A simple quantum circuit to implement the matrix exponentiation $e^{-i\frac{\theta}{2}Z\otimes Z}$. Other exponentials such as $e^{-i\frac{\theta}{2}X\otimes X}$ and $e^{-i\frac{\theta}{2}Y\otimes Y}$ can be implemented in a similar way modulo a change of basis.

$$|0\rangle\langle 0| \otimes I_j I_{j+1} + |1\rangle\langle 1| \otimes U_j U_{j+1} \tag{3.2}$$

can be implemented with two elementary controlled gates $|0\rangle\langle 0| \otimes I_j + |1\rangle\langle 1| \otimes U_j$ and $|0\rangle\langle 0| \otimes I_{j+1} + |1\rangle\langle 1| \otimes U_{j+1}$. The exponentiation of a Pauli string can be accomplished by a ladder circuit [83, Section 4.7.3] as illustrated in Figure 3.1.

3.2 Product-formula implementation details

We now describe the implementation details for the product-formula algorithm. As mentioned in Section 2.4, the key step is to choose a Trotter number r such that the simulation error is at most some desired ϵ. Here, we present *commutator bounds* that take advantage of the commutativity of Hamiltonian, tightening the previous analysis of Section 2.4.

Abstract commutator bounds. We recall some useful properties of Taylor expansion. For any $k \in \mathbb{N}$ and any analytic function $f\colon \mathbb{C} \to \mathbb{C}$ with $f(x) = \sum_{j=0}^{\infty} a_j x^j$, let $\mathcal{R}_k(f) := \sum_{j=k+1}^{\infty} a_j x^j$ denote the remainder of the Taylor series expansion of f up to order k.

Lemma 9. *If $\lambda \in \mathbb{C}$ and $H_1, \ldots, H_\Gamma$ are Hermitian, then*

$$\left\| \mathcal{R}_k \left(\prod_{\gamma=1}^{\Gamma} \exp(\lambda H_\gamma) \right) \right\| \leq \mathcal{R}_k \left(\exp \left(\sum_{\gamma=1}^{\Gamma} |\lambda| \cdot \|H_\gamma\| \right) \right). \tag{3.3}$$

Proof. We have

$$\prod_{\gamma=1}^{\Gamma} \exp(\lambda H_\gamma) = \prod_{\gamma=1}^{\Gamma} \sum_{j_\gamma=0}^{\infty} \lambda^{j_\gamma} \frac{H_\gamma^{j_\gamma}}{j_\gamma!} = \sum_{j_1,\ldots,j_\Gamma=0}^{\infty} \prod_{\gamma=1}^{\Gamma} \lambda^{j_\gamma} \frac{H_\gamma^{j_\gamma}}{j_\gamma!}, \tag{3.4}$$

so

$$\mathcal{R}_k \left(\prod_{\gamma=1}^{\Gamma} \exp(\lambda H_\gamma) \right) = \sum_{\substack{j_1,\ldots,j_\Gamma=0 \\ \sum_\gamma j_\gamma \geq k+1}}^{\infty} \prod_{\gamma=1}^{\Gamma} \lambda^{j_\gamma} \frac{H_\gamma^{j_\gamma}}{j_\gamma!}. \tag{3.5}$$

Using the triangle inequality and submultiplicativity of the norm, we find

$$\left\| \mathcal{R}_k \left(\prod_{\gamma=1}^{\Gamma} \exp(\lambda H_\gamma) \right) \right\| = \left\| \sum_{\substack{j_1,\ldots,j_\Gamma=0 \\ \sum_\gamma j_\gamma \geq k+1}}^{\infty} \prod_{\gamma=1}^{\Gamma} \lambda^{j_\gamma} \frac{H_\gamma^{j_\gamma}}{j_\gamma!} \right\| \leq \sum_{\substack{j_1,\ldots,j_\Gamma=0 \\ \sum_\gamma j_\gamma \geq k+1}}^{\infty} \prod_{\gamma=1}^{\Gamma} |\lambda|^{j_\gamma} \frac{\|H_\gamma\|^{j_\gamma}}{j_\gamma!}. \tag{3.6}$$

Finally, similarly as in (3.4) and (3.5), we have

$$\sum_{\substack{j_1,\ldots,j_\Gamma=0 \\ \sum_\gamma j_\gamma \geq k+1}}^{\infty} \prod_{\gamma=1}^{\Gamma} |\lambda|^{j_\gamma} \frac{\|H_\gamma\|^{j_\gamma}}{j_\gamma!} = \mathcal{R}_k \left(\prod_{\gamma=1}^{\Gamma} \exp(|\lambda| \cdot \|H_\gamma\|) \right) = \mathcal{R}_k \left(\exp \left(\sum_{\gamma=1}^{\Gamma} |\lambda| \cdot \|H_\gamma\| \right) \right), \tag{3.7}$$

which completes the proof. □

Lemma 10. *If $\lambda \in \mathbb{C}$, then $|\mathcal{R}_k(\exp(\lambda))| \leq \frac{|\lambda|^{k+1}}{(k+1)!} \exp(|\lambda|)$.*

Proof. Using the Taylor expansion of the exponential function, we have

$$|\mathcal{R}_k(\exp(\lambda))| = \left|\sum_{l=k+1}^{\infty} \frac{\lambda^l}{l!}\right| \leq \sum_{l=k+1}^{\infty} \frac{|\lambda|^l}{l!} = |\lambda|^{k+1} \sum_{l=k+1}^{\infty} \frac{|\lambda|^{l-(k+1)}}{l!} \tag{3.8}$$

$$= |\lambda|^{k+1} \sum_{l=0}^{\infty} \frac{|\lambda|^l}{(l+(k+1))!} \leq |\lambda|^{k+1} \sum_{l=0}^{\infty} \frac{|\lambda|^l}{l!} \frac{1}{(k+1)!} \tag{3.9}$$

$$= \frac{|\lambda|^{k+1}}{(k+1)!} \exp(|\lambda|), \tag{3.10}$$

which completes the proof. □

We now present improved error bounds for the first-, second-, and fourth-order product formula that exploits the commutation information of the Hamiltonian.

Theorem 11 (First-order commutator bound)**.** *Let $H_1, \ldots, H_\Gamma$ be Hermitian operators with norm at most $\Lambda := \max_\gamma \|H_\gamma\|$. Let $c := |\{(H_i, H_j) : [H_i, H_j] \neq 0,\ i < j\}|$ be the number of non-commuting pairs of operators, and let $t \in \mathbb{R}$. Then*

$$\left\| \exp\left(-it \sum_{\gamma=1}^{\Gamma} H_\gamma\right) - \left[\prod_{\gamma=1}^{\Gamma} \exp\left(-\frac{it}{r} H_\gamma\right)\right]^r \right\| \leq c\frac{(\Lambda t)^2}{r} + \frac{(\Gamma\Lambda|t|)^3}{3r^2} \exp\left(\frac{\Gamma\Lambda|t|}{r}\right). \tag{3.11}$$

Proof. We show that

$$\left\| \exp\left(\lambda \sum_{\gamma=1}^{\Gamma} H_\gamma\right) - \prod_{\gamma=1}^{\Gamma} \exp(\lambda H_\gamma) \right\| \leq C(|\lambda|\Lambda)^2 + \frac{(|\lambda|\Gamma\Lambda)^3}{3} \exp\left(|\lambda|\Gamma\Lambda\right), \tag{3.12}$$

which implies the claimed result by the triangle inequality. The upper bound (3.12)

can be established by explicitly computing the second-order error and bounding the higher-order errors by the norm of the remainder $\mathcal{R}_2$. The second-order error is bounded as follows:

$$\left\| \frac{\lambda^2}{2} \left(\sum_{\gamma=1}^{\Gamma} H_\gamma \right)^2 - \left[\frac{\lambda^2}{2} \sum_{\gamma=1}^{\Gamma} H_\gamma^2 + \lambda^2 \sum_{\gamma<\kappa} H_\gamma H_\kappa \right] \right\|$$

$$= \left\| \left[\frac{\lambda^2}{2} \sum_{\gamma=1}^{\Gamma} H_\gamma^2 + \frac{\lambda^2}{2} \sum_{\gamma<\kappa} H_\gamma H_\kappa + \frac{\lambda^2}{2} \sum_{\gamma>\kappa} H_\gamma H_\kappa \right] - \left[\frac{\lambda^2}{2} \sum_{\gamma=1}^{\Gamma} H_\gamma^2 + \lambda^2 \sum_{\gamma<\kappa} H_\gamma H_\kappa \right] \right\| \tag{3.13}$$

$$= \left\| \frac{\lambda^2}{2} \left[\sum_{\gamma>\kappa} H_\gamma H_\kappa - \sum_{\gamma<\kappa} H_\gamma H_\kappa \right] \right\| = \left\| \frac{\lambda^2}{2} \left[\sum_{\gamma>\kappa} H_\gamma H_\kappa - H_\kappa H_\gamma \right] \right\| \tag{3.14}$$

$$\leq c|\lambda|^2 \Lambda^2. \tag{3.15}$$

The rest of the proof proceeds similarly to the second half of the proof of Corollary 8; we omit the details. □

Theorem 12 (Second-order commutator bound)**.** *Let $H_1, \ldots, H_\Gamma$ be Hermitian operators with norm at most $\Lambda := \max_\gamma \|H_\gamma\|$, where each H_γ is a tensor product of Pauli operators. Define the augmented set of Hamiltonians*

$$\tilde{H}_\gamma = \begin{cases} H_\gamma, & 1 \leq \gamma \leq \Gamma \\ H_{2\Gamma-\gamma+1}, & \Gamma+1 \leq \gamma \leq 2\Gamma. \end{cases} \tag{3.16}$$

Let $f(\iota,\gamma) = 1$ *if* $\tilde{H}_\iota, \tilde{H}_\gamma$ *commute and* $f(\iota,\gamma) = -1$ *otherwise. Finally, let*

$$\Delta := |\{(\iota,\gamma) : f(\iota,\gamma) = -1, i \neq \gamma\}|, \tag{3.17}$$

$$\mathrm{T}_1 := |\{(\iota,\gamma,\kappa) : f(\iota,\gamma) = f(\gamma,\kappa) = f(\iota,\kappa) = 1,\ \iota < \gamma < \kappa\}|, \tag{3.18}$$

$$\begin{aligned}\mathrm{T}_2 := {} & |\{(\iota,\gamma,\kappa) : f(\iota,\gamma) = 1, f(\gamma,\kappa) = f(\iota,\kappa) = -1,\ \iota < \gamma < \kappa\}| \\ & + |\{(\iota,\gamma,\kappa) : f(\iota,\gamma) = f(\iota,\kappa) = -1,\ f(\gamma,\kappa) = 1,\ \iota < \gamma < \kappa\}|,\end{aligned} \tag{3.19}$$

$$\mathrm{T}_3 := |\{(\iota,\gamma,\kappa) : f(\iota,\gamma) = f(\gamma,\kappa) = -1,\ f(\iota,\kappa) = 1,\ \iota < \gamma < \kappa\}|, \tag{3.20}$$

$$\mathrm{T}_4 := |\{(\iota,\gamma,\kappa) : \textit{all other cases}\}| \tag{3.21}$$

where $\iota,\gamma,\kappa \in \{1,\ldots,2\Gamma\}$*, and let* $t \in \mathbb{R}$*. Then*

$$\begin{aligned}&\left\| \exp\left(-it\sum_{\gamma=1}^{\Gamma} H_\gamma\right) - \left[\mathscr{S}_2\left(-\frac{it}{r}\right)\right]^r \right\| \\ &\quad \leq \frac{\Lambda^3|t|^3}{r^2}\left\{\frac{1}{24}\Delta + \frac{1}{12}\mathrm{T}_2 + \frac{1}{6}\mathrm{T}_3 + \frac{1}{8}\mathrm{T}_4\right\} + \frac{4(\Gamma\Lambda t)^4}{3r^3}\exp\left(\frac{2\Gamma\Lambda|t|}{r}\right).\end{aligned} \tag{3.22}$$

Proof. As in the proof of Theorem 11, we explicitly compute the third-order error and bound the higher-order terms by $\mathcal{R}_3$. First we show that the first-order formula

$$\left\| \exp\left(\lambda\sum_{\gamma=1}^{\Gamma} H_\gamma\right) - \prod_{\gamma=1}^{\Gamma}\exp(\lambda H_\gamma) \right\| \tag{3.23}$$

has a third-order error of at most

$$|\lambda|^3\Lambda^3\left(\frac{1}{3}\bar{\Delta} + \frac{2}{3}\bar{\mathrm{T}}_2 + \frac{4}{3}\bar{\mathrm{T}}_3 + \bar{\mathrm{T}}_4\right), \tag{3.24}$$

where the coefficients $\bar{\Delta}, \bar{\mathrm{T}}_2, \bar{\mathrm{T}}_3, \bar{\mathrm{T}}_4$ are defined as in (3.17)–(3.21), but with respect to the original Hamiltonians $\{H_\gamma\}_{\gamma=1}^{\Gamma}$ instead of $\{\tilde{H}_\gamma\}_{\gamma=1}^{2\Gamma}$.

The third-order term in $\exp(\lambda \sum_{\gamma=1}^{\Gamma} H_\gamma)$ is

$$\frac{1}{3!}\left(\lambda \sum_{\gamma=1}^{\Gamma} H_\gamma\right)^3 = \frac{\lambda^3}{6} \sum_{\iota,\gamma,\kappa} H_\iota H_\gamma H_\kappa, \tag{3.25}$$

whereas the third-order term in $\prod_{\gamma=1}^{\Gamma} \exp(\lambda H_\gamma)$ is

$$\begin{aligned} &\frac{\lambda^3}{6} \sum_{\iota} H_\iota^3 + \frac{\lambda^3}{2} \sum_{\iota<\kappa} H_\iota^2 H_\kappa + \frac{\lambda^3}{2} \sum_{\iota<\kappa} H_\iota H_\kappa^2 + \lambda^3 \sum_{\iota<\gamma<\kappa} H_\iota H_\gamma H_\kappa \\ &\quad = \frac{\lambda^3}{6} \sum_{\iota} H_\iota^3 + \frac{\lambda^3}{2} \sum_{\iota\neq k} H_\iota^2 H_\kappa + \lambda^3 \sum_{\iota<\gamma<\kappa} H_\iota H_\gamma H_\kappa, \end{aligned} \tag{3.26}$$

where we have used the fact that the square of any Pauli operator is the identity. Taking the difference gives

$$\frac{\lambda^3}{6} \sum_{\iota\neq\gamma} H_\iota [H_\gamma, H_\iota] + \frac{\lambda^3}{6} \sum_{\substack{\iota,\gamma,\kappa \\ \text{pairwise different}}} H_\iota H_\gamma H_\kappa - \lambda^3 \sum_{\iota<\gamma<\kappa} H_\iota H_\gamma H_\kappa. \tag{3.27}$$

The norm of the first term is at most

$$\frac{1}{3}|\lambda|^3 \Lambda^3 \bar{\Delta}, \tag{3.28}$$

whereas the last two terms can be written as follows:

$$\begin{aligned}
&\frac{\lambda^3}{6} \sum_{\substack{\iota_1,\iota_2,\iota_3 \\ \text{pairwise different}}} H_{\iota_1} H_{\iota_2} H_{\iota_3} - \lambda^3 \sum_{\iota_1<\iota_2<\iota_3} H_{\iota_1} H_{\iota_2} H_{\iota_3} \\
&= \frac{\lambda^3}{6} \sum_{\iota_1<\iota_2<\iota_3} \sum_{\sigma \in \mathrm{Sym}(3)} H_{\iota_{\sigma(1)}} H_{\iota_{\sigma(2)}} H_{\iota_{\sigma(3)}} - \lambda^3 \sum_{\iota_1<\iota_2<\iota_3} H_{\iota_1} H_{\iota_2} H_{\iota_3} \qquad (3.29) \\
&= \frac{\lambda^3}{6} \sum_{\iota_1<\iota_2<\iota_3} \big(1 + f(1,2) + f(2,3) + f(1,2)f(1,3)f(2,3) \qquad (3.30) \\
&\quad + f(1,3)f(1,2) + f(1,3)f(2,3)\big) H_{\iota_1} H_{\iota_2} H_{\iota_3} - \lambda^3 \sum_{\iota_1<\iota_2<\iota_3} H_{\iota_1} H_{\iota_2} H_{\iota_3}. \qquad (3.31)
\end{aligned}$$

(Here Sym(3) denotes the symmetric group on three elements.) By performing case analysis, we can evaluate the coefficients and upper bound the norm by

$$|\lambda|^3 \Lambda^3 \left(\frac{2}{3} \bar{\mathrm{T}}_2 + \frac{4}{3} \bar{\mathrm{T}}_3 + \bar{\mathrm{T}}_4 \right). \qquad (3.32)$$

Combining (3.28) and (3.32), we obtain the claimed upper bound (3.24) for the third-order error in the first-order formula.

Now we consider the second-order formula. Similarly to the proof of Theorem 11, we begin by proving the bound

$$\left\| \exp\left(\lambda \sum_{\gamma=1}^{\Gamma} H_\gamma \right) - \mathscr{S}_2(\lambda) \right\| \leq (|\lambda|\Lambda)^3 \left(\frac{\Delta}{24} + \frac{\mathrm{T}_2}{12} + \frac{\mathrm{T}_3}{6} + \frac{\mathrm{T}_4}{8} \right) + \frac{4}{3} (\Gamma |\lambda| \Lambda)^4 \exp(2\Gamma |\lambda| \Lambda), \qquad (3.33)$$

which implies (3.22) by the triangle inequality.

To establish (3.33), we apply (3.24) to the augmented Hamiltonian list $\{\tilde{H}_\gamma\}_{\gamma=1}^{2\Gamma}$

with λ replaced by $\frac{\lambda}{2}$. This shows that the third-order error is at most

$$\frac{|\lambda|^3\Lambda^3}{8}\left(\frac{\Delta}{3}+\frac{2\mathrm{T}_2}{3}+\frac{4\mathrm{T}_3}{3}+\mathrm{T}_4\right). \tag{3.34}$$

The higher-order errors can be bounded by a routine calculation as

$$\mathcal{R}_3\left(\exp\left(\lambda\sum_{\gamma=1}^{\Gamma}H_\gamma\right)-\mathscr{S}_2(\lambda)\right)\leq 2\mathcal{R}_3(\exp(2\Gamma|\lambda|\Lambda))\leq 2\frac{(2\Gamma|\lambda|\Lambda)^4}{4!}\exp(2\Gamma|\lambda|\Lambda). \tag{3.35}$$

This completes the proof of (3.33). The remainder of the proof proceeds similarly to the second half of the proof of Corollary 8. □

A similar bound holds for the fourth-order formula, as follows.

Theorem 13 (Fourth-order commutator bound)**.** *Let $H_1,\ldots,H_\Gamma$ be Hermitian operators with norm at most $\Lambda := \max_j \|H_j\|$, where each H_j is a tensor product of Pauli operators. Define the augmented set of Hamiltonians*

$$\tilde{H}_j=\begin{cases}H_{j-2h\Gamma}, & 2h\Gamma+1\leq j\leq(2h+1)\Gamma\\ H_{2(h+1)\Gamma-j+1}, & (2h+1)\Gamma+1\leq j\leq 2(h+1)\Gamma\end{cases}\qquad h\in\{0,1,2,3,4\}. \tag{3.36}$$

Let $f(i,j) = 1$ *if* $\tilde{H}_i, \tilde{H}_j$ *commute and* $f(i,j) = -1$ *otherwise. Finally, let*

$$n_a := |\{(i,j) : f(i,j) = a_1,\ i < j\}| \tag{3.37}$$

$$n_{b_1b_2b_3} := |\{(i,j,k) : f(i,j) = b_1,\ f(i,k) = b_2,\ f(j,k) = b_3,\ i < j < k\}| \tag{3.38}$$

$$\begin{aligned} n_{c_1\ldots c_6} := |\{(i,j,k,l) : f(i,j) = c_1,\ f(i,k) = c_2,\ f(i,l) = c_3,\\ f(j,k) = c_4,\ f(j,l) = c_5,\ f(k,l) = c_6,\ i < j < k < l\}| \end{aligned} \tag{3.39}$$

$$\begin{aligned} n_{d_1\ldots d_{10}} := |\{(i,j,k,l,m) : f(i,j) = d_1,\ f(i,k) = d_2,\ f(i,l) = d_3,\\ f(i,m) = d_4,\ f(j,k) = d_5,\ f(j,l) = d_6,\ f(j,m) = d_7,\ f(k,l) = d_8,\\ f(k,m) = d_9,\ f(l,m) = d_{10},\ i < j < k < l < m\}| \end{aligned} \tag{3.40}$$

where $i,j,k,l,m \in \{1,\ldots,10\Gamma\}$*, let* $t \in \mathbb{R}$*, and let* $p := 1/(4 - 4^{1/3})$*. Then*

$$\left\| \exp\left(-it\sum_{j=1}^{\Gamma} H_j\right) - \left[\mathscr{S}_4\left(-\frac{it}{r}\right)\right]^r \right\| \tag{3.41}$$

$$\begin{aligned} &\leq \left(\frac{4p-1}{2}\Lambda|t|\right)^5 \frac{1}{5!r^4}\Bigg\{ \sum_{a\in\pm 1} c_a n_a + \sum_{b_1,b_2,b_3\in\pm 1} c_{b_1b_2b_3} n_{b_1b_2b_3} \\ &\quad + \sum_{c_1,\ldots,c_6\in\pm 1} c_{c_1\ldots c_6} n_{c_1\ldots c_6} + \sum_{d_1,\ldots,d_{10}\in\pm 1} c_{d_1\ldots d_{10}} n_{d_1\ldots d_{10}} \Bigg\} \\ &\quad + 2\frac{(5(4p-1)\Gamma\Lambda t)^6}{6!\cdot r^5} \exp\left(\frac{(5(4p-1)\Gamma\Lambda|t|}{r}\right) \end{aligned} \tag{3.42}$$

for some real coefficients c_a*,* $c_{b_1b_2b_3}$*,* $c_{c_1\ldots c_6}$*,* $c_{d_1\ldots d_{10}}$*.*

We omit the proof, which proceeds along similar lines to that of Theorem 12. Note that similar bounds also hold for higher-order formulas, although the analysis

becomes more involved.

The coefficients c_a, $c_{b_1b_2b_3}$, $c_{c_1\ldots c_6}$, $c_{d_1\ldots d_{10}}$ can in principle be determined by a computer program. To illustrate the idea, we show how to determine the coefficient c_{-1} in (3.41). Similar arguments can be used to determine all the coefficients in the bound. However, the list of coefficients is long, so we omit it here.

First consider the fifth-order terms of the expression

$$\exp\left(\sum_{j=1}^{\Gamma} H_j\lambda\right) - \exp(H_1\lambda)\cdots\exp(H_\Gamma\lambda). \tag{3.43}$$

The coefficient c_{-1} of n_{-1} counts the pairs of non-commuting terms H_i and H_j. The second term in (3.43) contributes

$$\frac{\lambda^4}{4!}\lambda\sum_{i<j}\left(H_i^4H_j + H_iH_j^4\right) + \frac{\lambda^3}{3!}\frac{\lambda^2}{2!}\sum_{i<j}\left(H_i^3H_j^2 + H_i^2H_j^3\right), \tag{3.44}$$

whereas the first term in (3.43) contributes

$$\begin{aligned}
&\frac{\lambda^5}{5!}\sum_{i\neq j}\left(H_i^4H_j + H_i^3H_jH_i + H_i^2H_jH_i^2 + H_iH_jH_i^3 + H_jH_i^4\right)\\
+&\frac{\lambda^5}{5!}\sum_{i\neq j}\left(H_i^3H_j^2 + H_i^2H_jH_iH_j + H_i^2H_j^2H_i + H_iH_jH_i^2H_j + H_iH_jH_iH_jH_i\right.\\
&\qquad\left. + H_iH_j^2H_i^2 + H_jH_i^3H_j + H_jH_i^2H_jH_i + H_jH_iH_jH_i^2 + H_j^2H_i^3\right).
\end{aligned} \tag{3.45}$$

Under the assumption that the terms of the Hamiltonian are tensor products of Pauli operators, we can interchange the order of multiplication, possibly introduc-

ing minus signs. Thus (3.45) equals

$$\frac{\lambda^5}{5!}\sum_{i<j}\big(\big(3+2f(i,j)\big)H_i^4H_j+\big(3+2f(i,j)\big)H_iH_j^4\big)$$
$$+\frac{\lambda^5}{5!}\sum_{i<j}\big(\big(6+4f(i,j)\big)H_i^3H_j^2+\big(6+4f(i,j)\big)H_i^2H_j^3\big)\,. \tag{3.46}$$

Subtracting (3.46) from (3.44) and setting $f(i,j)=-1$, we find

$$\frac{\lambda^5}{5!}\sum_{i<j}\big(4H_i^4H_j+4H_iH_j^4+8H_i^3H_j^2+8H_i^2H_j^3\big)\,, \tag{3.47}$$

whose spectral norm is bounded by

$$\frac{|\lambda|^5}{5!}24\Lambda^5N_{-1}. \tag{3.48}$$

Comparing the result to (3.41), we find that $c_{-1}=24$.

Concrete commutator bounds. To apply the above commutator bounds, we must compute the number of tuples of terms in the Hamiltonian satisfying certain commutation relations (e.g., equations (3.37)–(3.40) for the fourth-order bound). While this can be done in polynomial time provided the degree is constant, a direct approach is prohibitive in practice.

However, for the Hamiltonian (3.1), it is possible to show that each number of tuples is given by a low-degree polynomial in n. In turn, this means that the lowest-order contribution to the error is also a polynomial in n. Thus, by

performing polynomial interpolation on a constant number of numerically-obtained values, we can determine a closed-form expression for general n. We give concrete commutator bounds for the second- and fourth-order formulas whose proof can be found in [34, Appendix F.2].

Theorem 14 (Second-order commutator bound, succinct form)**.** *Let H be the Hamiltonian (3.1), with terms ordered as*

$$X_1X_2, \ldots, X_{n-1}X_n, X_nX_1,\ Y_1Y_2, \ldots, Y_{n-1}Y_n, Y_nY_1,\ Z_1Z_2, \ldots, Z_{n-1}Z_n, Z_nZ_1,$$
$$Z_1, \ldots, Z_n. \tag{3.49}$$

Then the error in the second-order product formula approximation satisfies

$$\|\exp(-iHt) - [\mathscr{S}_2(-it/r)]^r\| \leq \frac{\Lambda^3|t|^3}{r^2}\mathrm{T}_2(n) + \frac{4(4n\Lambda t)^4}{3r^3}\exp\left(\frac{8n\Lambda|t|}{r}\right), \tag{3.50}$$

where

$$\mathrm{T}_2(n) := \begin{cases} 194, & n = 3 \\ 40n^2 - 58n, & n \geq 4. \end{cases} \tag{3.51}$$

Theorem 15 (Fourth-order commutator bound, succinct form)**.** *Let H be the Hamiltonian (3.1), with terms ordered as in (3.49), and let $p := 1/(4 - 4^{1/3})$.*

Then the error in the fourth-order product formula approximation satisfies

$$\begin{aligned}&\|\exp(-iHt) - [\mathscr{S}_4(-it/r)]^r\| \\ &\quad \leq \left(\frac{4p-1}{2}\Lambda|t|\right)^5 \frac{1}{5!\cdot r^4}\mathrm{T}_4(n) + 2\frac{(20(4p-1)n\Lambda t)^6}{6!\cdot r^5}\exp\left(\frac{20(4p-1)n\Lambda|t|}{r}\right),\end{aligned} \tag{3.52}$$

where

$$\mathrm{T}_4(n) := \begin{cases} 23073564672, & n = 3 \\ 94192316416, & n = 4 \\ 278878851840, & n = 5 \\ 1280000000n^4 - 7701760000n^3 + 23685120000n^2 - 30224677632n, & n \geq 6. \end{cases} \tag{3.53}$$

We now consider the asymptotic gate complexity of the product-formula algorithm using our commutator bounds. Take the fourth-order bound as an example. With $\Lambda = \Theta(1)$ and $t = \Theta(n)$, the commutator bound is

$$\|\exp(-iHt) - [\mathscr{S}_4(-it/r)]^r\| \leq \mathcal{O}\left(\frac{n^9}{r^4} + \frac{n^{12}}{r^5}\right). \tag{3.54}$$

To guarantee that the simulation error ϵ is at most some constant, it suffices to use $r = \mathcal{O}(n^{2.4})$ segments. Since the circuit for each segment has size $\mathcal{O}(n)$, we have an overall gate complexity of $\mathcal{O}(n^{3.4})$. Along similar lines, we find gate complexities of

$\mathcal{O}(n^4)$ (resp., $\mathcal{O}(n^{11/3})$) for the first-order (resp., second-order) commutator bound. These bounds improve the asymptotic gate complexities of the product-formula algorithm with 1-norm scaling (as established in Corollary 8), which give $\mathcal{O}(n^5)$ for the first-order formula, $\mathcal{O}(n^4)$ for second order, and $\mathcal{O}(n^{3.5})$ for fourth order.

We only present concrete commutator bounds for the first-, second-, and fourth-order product formulas. In general, to obtain the $2k$th-order commutator bound, one must count the number of $(2k+1)$-tuples satisfying a certain commutation pattern in a list of operators of length $8 \cdot 5^{k-1} n$. For $k \geq 3$, computing the exact form of the $(2k)$th order bound seems challenging even with the help of polynomial interpolation.

Nevertheless, it is still possible to obtain the asymptotic n-dependence of the commutator bound. The key step is to study those $(2k+1)$-tuples for which all pairs of operators commute with each other. The number n_{comm} of such commuting $(2k+1)$-tuples satifsfies

$$(8 \cdot 5^{k-1})^{2k+1} \binom{n-2k}{2k+1} \leq n_{\text{comm}} \leq \binom{8 \cdot 5^{k-1} n}{2k+1}. \tag{3.55}$$

We thus conclude that N_{comm} is a polynomial in n whose leading term is $(8 \cdot 5^{k-1} n)^{2k+1}$.

When we Taylor expand the evolutions $\exp(\lambda H)$ and $\mathscr{S}_{2k}(\lambda)$, those $(2k+1)$-tuples for which all pairs of operators commute with each other cancel. The remaining terms are either $(2k+1)$-tuples where at least one pair of operators do

not commute, or l-tuples with $l \leq 2k$. Our above discussion shows that there are $\mathcal{O}(n^{2k})$ such tuples. Therefore, the $2k$th-order commutator bound takes the form

$$\begin{aligned} \|\exp(-iHt) - [\mathscr{S}_{2k}(-it/r)]^r\| &\leq \mathcal{O}\left(\frac{|t|^{2k+1}n^{2k}}{r^{2k}} + \frac{(nt)^{2k+2}}{r^{2k+1}}\right) \\ &= \mathcal{O}\left(\frac{n^{4k+1}}{r^{2k}} + \frac{n^{4k+4}}{r^{2k+1}}\right). \end{aligned} \tag{3.56}$$

To ensure that the simulation error is $\mathcal{O}(1)$ for $t = n$, it suffices to choose $r = \mathcal{O}(n^{2+2/(2k+1)})$, which leads to a total gate complexity of $\mathcal{O}(n^{3+2/(2k+1)})$. This improves over the 1-norm scaled bound (Corollary 8), which give complexity $\mathcal{O}(n^{3+1/k})$.

Empirical bounds. While the above bounds provide rigorous correctness guarantees, they can be very loose. To understand the minimum resources that suffice for product formula simulation, we estimate their empirical performance. Of course, since quantum simulation is computationally challenging, we can only directly compute the actual simulation error for small instances. Using binary search, we find the value of r (the total number of segments) that just suffices to ensure error 10^{-3}. We extrapolate this behavior to produce a non-rigorous estimate of the performance of product formula simulation for instances of arbitrary size. We emphasize that the resulting *empirical bound* does not come with a guarantee of correctness. Nevertheless, we believe it better captures the true performance of product formula simulations and indicates the extent to which our rigorous bounds are loose.

We numerically simulate the product formula algorithm for systems of size 5

to 12, determining the value of r required to ensure error 10^{-3} as described above, and averaging over five random choices of the local field strengths h_j. We fit these data to power laws, as depicted in Figure 3.2. We find

$$r_1 = 2417n^{1.964},\ r_2 = 39.47n^{1.883},\ r_4 = 4.035n^{1.555},\ r_6 = 1.789n^{1.311},\ r_8 = 1.144n^{1.141}, \tag{3.57}$$

where r_i is the number of segments for the ith-order formula to produce a simulation that is accurate to within 10^{-3}. Considering the size of the circuit for each step, this suggests an asymptotic complexity of roughly $9668n^{2.964}$ for the first-order formula, $315.8n^{2.883}$ for second order, $161.4n^{2.555}$ for fourth order, $357.8n^{2.311}$ for sixth order, and $1144n^{2.141}$ for eighth order.

3.3 Taylor-series implementation details

In this section, we discuss technical details related to the implementation of the Taylor-series algorithm as introduced in Section 2.5. We first present concrete error bounds to determine the truncate order K. We then describe how to implement select(V), a major component of the algorithm.

Error analysis. We begin by bounding the error introduced by truncating the Taylor series.

Lemma 16. *With the definitions of* $\tilde{\mathscr{U}}(t)$ *in (2.26) and* $t_{\text{seg}} := \ln 2 / \|\alpha\|_1$, *we have*

$$\left\| \tilde{\mathscr{U}}(t) - \exp(-iHt) \right\| \leq 2\frac{(\ln 2)^{\mathrm{K}+1}}{(\mathrm{K}+1)!} \tag{3.58}$$

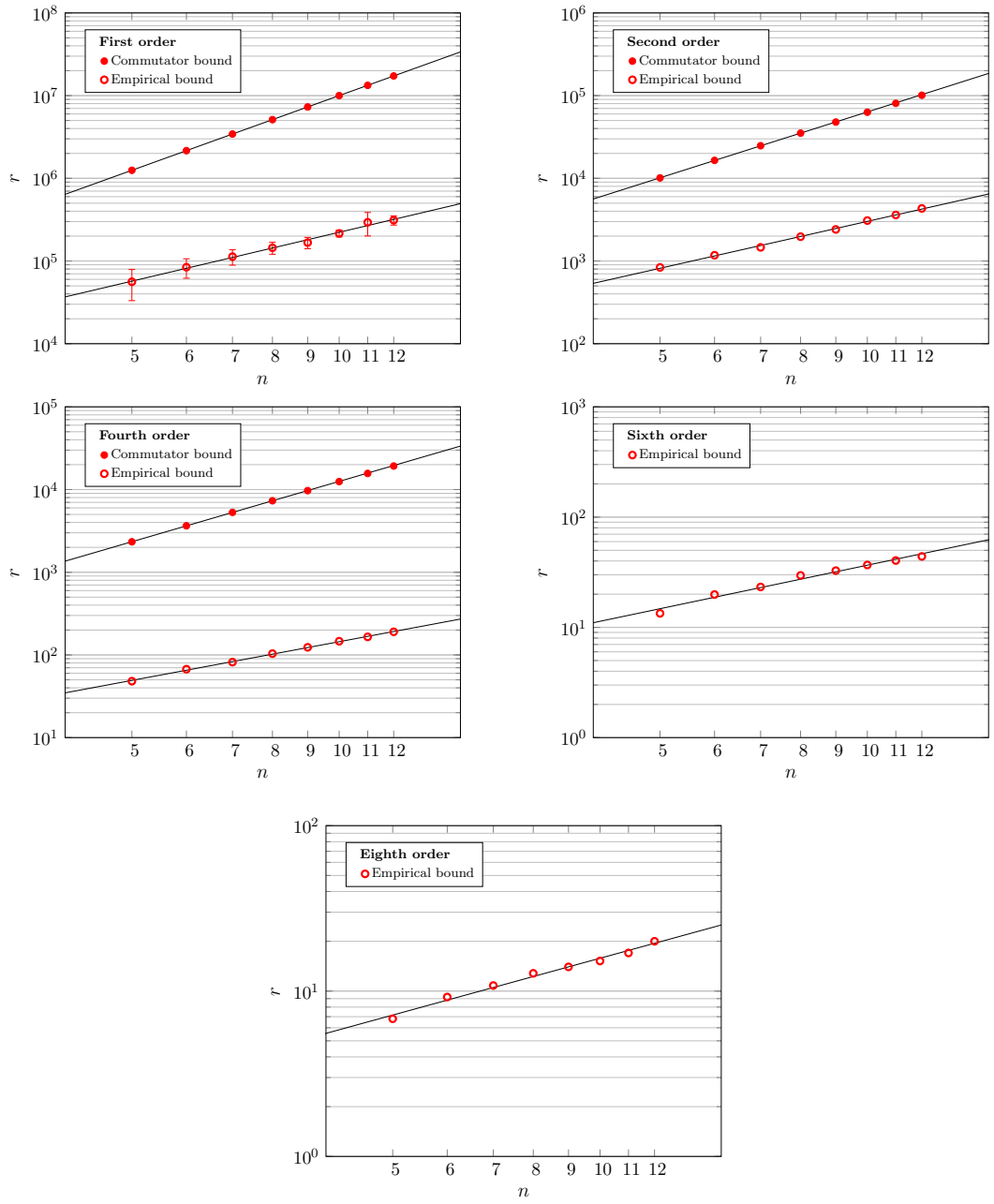

Figure 3.2: Comparison of the values of r using the commutator and empirical bounds for formulas of order 1, 2, and 4, and values of r for the empirical bound for formulas of order 6 and 8. Straight lines show power-law fits to the data. The error bars for product formulas of order greater than 1 are negligibly small, so we omit them from the plots.

for any $t \leq t_{\text{seg}}$.

Proof. We have

$$\left\| \tilde{\mathscr{U}}(t) - \exp(-iHt) \right\| = \left\| \sum_{\kappa=\mathrm{K}+1}^{\infty} \frac{(-iHt)^\kappa}{\kappa!} \right\| \tag{3.59}$$

$$\leq \sum_{\kappa=\mathrm{K}+1}^{\infty} \frac{(\|H\|\, t)^\kappa}{\kappa!} \tag{3.60}$$

$$\leq \sum_{\kappa=\mathrm{K}+1}^{\infty} \frac{((\alpha_1 + \cdots + \alpha_\Gamma)t)^\kappa}{\kappa!} \tag{3.61}$$

$$\leq \sum_{\kappa=\mathrm{K}+1}^{\infty} \frac{(\ln 2)^\kappa}{\kappa!} \tag{3.62}$$

$$\leq 2\frac{(\ln 2)^{\mathrm{K}+1}}{(\mathrm{K}+1)!}, \tag{3.63}$$

where the final inequality follows from Lemma 10. □

Next we consider the error induced by the isometry $\mathscr{V}(t) := -WRW^\dagger RW(|0\rangle \otimes I)$ as in (2.34). It is straightforward to verify that

$$(\langle 0| \otimes I)\mathscr{V}(t) = \frac{3}{s}\tilde{\mathscr{U}}(t) - \frac{4}{s^3}\tilde{\mathscr{U}}(t)\tilde{\mathscr{U}}(t)^\dagger\tilde{\mathscr{U}}(t). \tag{3.64}$$

As discussed in Section 2.5, we rotate an ancilla qubit to increase the value of s to be precisely $s = 2$. Then the following bound characterizes the error in the implementation of the Taylor series.

Lemma 17. *Suppose* $\left\|\tilde{U} - U\right\| \leq \delta$ *for some unitary operator* U*. Then*

$$\left\|\frac{3}{2}\tilde{U} - \frac{1}{2}\tilde{U}\tilde{U}^\dagger\tilde{U} - U\right\| \leq \frac{\delta^2 + 3\delta + 4}{2}\delta. \tag{3.65}$$

Proof. Since $\left\|\tilde{U} - U\right\| \leq \delta$, we have

$$\left\|\tilde{U}\right\| \leq \left\|\tilde{U} - U\right\| + \|U\| \leq 1 + \delta \tag{3.66}$$

and therefore

$$\left\|\tilde{U}\tilde{U}^\dagger - I\right\| \leq \left\|(\tilde{U} - U)\tilde{U}^\dagger\right\| + \left\|U(\tilde{U}^\dagger - U^\dagger)\right\| \leq \delta(2 + \delta). \tag{3.67}$$

Thus, by the triangle inequality, we have

$$\left\|\frac{3}{2}\tilde{U} - \frac{1}{2}\tilde{U}\tilde{U}^\dagger\tilde{U} - U\right\| \leq \left\|\tilde{U} - U\right\| + \frac{1}{2}\left\|\tilde{U} - \tilde{U}\tilde{U}^\dagger\tilde{U}\right\| \tag{3.68}$$

$$\leq \delta + \frac{\delta(2 + \delta)(1 + \delta)}{2} \tag{3.69}$$

$$= \frac{\delta^2 + 3\delta + 4}{2}\delta \tag{3.70}$$

as claimed. □

We use the following basic property of contractions (operators of norm at most 1), which is easily proved using the triangle inequality.

Lemma 18. *Suppose operators* U_i *and* V_i *are contractions for all* $i \in \{1, \ldots, r\}$*.*

If $\|U_i - V_i\| \leq \eta$ *for all* i*, then*

$$\|U_r \cdots U_2 U_1 - V_r \cdots V_2 V_1\| \leq r\eta. \tag{3.71}$$

We also use the following lemma, which bounds the error introduced by normalization.

Lemma 19. *Suppose* $\||\phi\rangle\| = 1$*,* $\||\psi\rangle\| \leq 1$*, and* $\||\psi\rangle - |\phi\rangle\| \leq \xi < 1$*. Then*

$$\left\| \frac{|\psi\rangle}{\||\psi\rangle\|} - |\phi\rangle \right\| \leq \sqrt{1+\xi} - \sqrt{1-\xi}. \tag{3.72}$$

Proof. Decompose $|\psi\rangle$ as

$$|\psi\rangle = \alpha|\phi\rangle + \beta|\phi^\perp\rangle \tag{3.73}$$

for some normalized state $\left|\phi^\perp\right\rangle$ orthogonal to $|\phi\rangle$. Clearly $|\alpha|^2 + |\beta|^2 \leq 1$ since $\||\psi\rangle\| \leq 1$. Furthermore, the assumption $\||\psi\rangle - |\phi\rangle\| \leq \xi$ implies

$$|\alpha - 1|^2 + |\beta|^2 \leq \xi^2, \tag{3.74}$$

so

$$|\alpha|^2 + |\beta|^2 \leq \xi^2 + 2\Re(\alpha) - 1 \tag{3.75}$$

with the real part $1 - \xi \leq \Re(\alpha) \leq 1 + \xi$.

Then we have

$$\left\| \frac{|\psi\rangle}{\||\psi\rangle\|} - |\phi\rangle \right\| = \sqrt{2 - \frac{2\Re(\alpha)}{\sqrt{|\alpha|^2 + |\beta|^2}}} \tag{3.76}$$

$$\leq \sqrt{2 - \frac{2\Re(\alpha)}{\sqrt{\xi^2 + 2\Re(\alpha) - 1}}} \tag{3.77}$$

$$\leq \sqrt{2 - \frac{2(1-\xi^2)}{\sqrt{\xi^2 + 2(1-\xi^2) - 1}}} \tag{3.78}$$

$$= \sqrt{1+\xi} - \sqrt{1-\xi}. \tag{3.79}$$

Here the first inequality uses (3.75) and the fact that $\Re(\alpha) \geq 1 - \xi \geq 0$, and the second inequality follows since the function $x/\sqrt{2x - 1 + \xi^2}$ attains its minimum at $x = 1 - \xi^2$ within the interval $1 - \xi \leq x \leq 1 + \xi$. □

With all the above lemmas in hand, we are ready to prove an explicit error bound for the Taylor-series algorithm.

Theorem 20. *The Taylor-series algorithm achieves error at most* $\sqrt{1+\xi} - \sqrt{1-\xi}$ *with success probability at least* $(1-\xi)^2$*, where*

$$\xi = r\frac{\delta^2 + 3\delta + 4}{2}\delta \quad \textit{with} \quad \delta = 2\frac{(\ln 2)^{\mathrm{K}+1}}{(\mathrm{K}+1)!}. \tag{3.80}$$

Proof. For $t \in \{t_{\mathrm{seg}}, t_{\mathrm{rem}}\}$, Lemma 16 shows that

$$\left\| \tilde{U}(t) - \exp(-iHt) \right\| \leq \delta, \tag{3.81}$$

and Lemma 17 shows that

$$\|(\langle 0| \otimes I)\mathscr{V}(t) - \exp(-iHt)\| \leq \xi/r. \tag{3.82}$$

Since $\mathscr{V}(t)$ is an isometry, $(\langle 0| \otimes I)\mathscr{V}(t)$ is a contraction, so by Lemma 18, we have

$$\left\|(\langle 0| \otimes I)\mathscr{V}(t_{\text{rem}})\big((\langle 0| \otimes I)\mathscr{V}(t_{\text{seg}})\big)^{r-1} - \exp(-itH)\right\| \leq \xi. \tag{3.83}$$

The claim about the success probability follows by applying the triangle inequality, and the accuracy can be established by invoking Lemma 19. □

To apply this bound, we must determine the truncation order K that achieves the desired error bound ϵ. Just as for the product formula error bounds presented in Section 3.2, it does not seem possible to compute K in closed form. However, since K can only take integer values, it is straightforward to tabulate the error estimates corresponding to all potentially relevant values of K, as shown in Table 3.1. Using the known value of r, we can then determine which value of K suffices to ensure small error.

In Section 3.2, we presented empirical error bounds for simulations based on product formulas. It would be natural to perform a similar analysis of the error in the Taylor-series algorithm. Unfortunately, it is intractable to find an empirical bound by direct simulation since the number of ancilla qubits used by the Taylor-series algorithm puts it beyond the reach of classical simulation even for very small

K	ξ/r
1	1.3626
2	0.24118
3	0.039031
4	0.0053441
5	0.00061628
6	6.10123×10^{-5}
7	5.28621×10^{-6}
8	4.07124×10^{-7}
9	2.821965×10^{-8}
10	1.778215×10^{-9}

Table 3.1: Lookup table for the truncation order K, with s boosted to be 2 in each segment.

systems. A more limited alternative would be to use empirical data to improve the estimated error of the truncated Taylor series. However, since

$$\frac{(\ln 2)^{\mathrm{K}+1}}{(\mathrm{K}+1)!} \leq \sum_{\kappa=\mathrm{K}+1}^{\infty} \frac{(\ln 2)^{\kappa}}{\kappa!} \leq 2\frac{(\ln 2)^{\mathrm{K}+1}}{(\mathrm{K}+1)!}, \tag{3.84}$$

the estimated error can be improved by a factor of at most 2, which results in an additive offset of at most $\ln 2$ for the truncation order K. Thus we do not consider such a bound in our analysis.

Implementation of select(V)**.** A crucial step in the implementation of the Taylor-series algorithm (and in the quantum-signal-processing algorithm) is to synthesize the select(V) gates. The cost of this implementation depends strongly on the chosen representation for the control register.

Recall from (2.30) that the select(V) operation has the form

$$\text{select}(V) := \sum_{\xi=0}^{\Xi-1} |\xi\rangle\langle\xi| \otimes \tilde{V}_\xi. \tag{3.85}$$

For the Taylor-series algorithm, the operators $\tilde{V}_\xi$ are defined via (2.29). Here the index ξ labels a value $\kappa \in \{0, 1, \ldots, \mathrm{K}\}$ and indices $\gamma_1, \ldots, \gamma_\kappa \in \{1, \ldots, \Gamma\}$. Perhaps the most straightforward approach is to represent the entire control register with a binary encoding using $\log_2(\mathrm{K}+1) + \mathrm{K}\log_2\Gamma$ bits. However, as pointed out in [14], we can significantly reduce the gate complexity by choosing a different encoding of the control register.

Specifically, we use a unary encoding to label κ and a binary encoding for each $\gamma_1, \ldots, \gamma_\kappa$. With such an encoding, the instance of select(V) in the Taylor-series algorithm can be represented as the map

$$|1^\kappa 0^{\mathrm{K}-\kappa}\rangle|\gamma_1, \ldots, \gamma_\mathrm{K}\rangle|\psi\rangle \mapsto |1^\kappa 0^{\mathrm{K}-\kappa}\rangle|\gamma_1, \ldots, \gamma_\mathrm{K}\rangle(-i)^\kappa H_{\gamma_1}\cdots H_{\gamma_\kappa}|\psi\rangle. \tag{3.86}$$

We implement this transformation as follows. Conditioned on the jth qubit of the unary encoding of κ being 1, and the jth coordinate of $\gamma_1, \ldots, \gamma_\mathrm{K}$ being the binary encoding of γ_j, we apply $(-i)H_{\gamma_j}$. Compared to an entirely binary encoding, this approach only requires an additional $\lceil \mathrm{K}+1-\log_2(\mathrm{K}+1)\rceil$ qubits, which is a modest increase since K is typically small (see Table 3.1). In return, instead of selecting on a large register of $\Theta(\mathrm{K}\log\Gamma)$ bits, we can perform $\mathrm{K}+1$ independent selections

on registers of $\log_2 \Gamma$ bits, each controlled by a single qubit.

To implement a single select operation of the form $\sum_{\gamma=0}^{\Gamma-1} |\gamma\rangle \langle\gamma| \otimes U_\gamma$, we would need to cycle the value of a designated ancilla qubit through Γ Boolean products of ω literals, where in each of the products, each of the variables $x_1, \ldots, x_\omega$ appears exactly once (either negated or not). We then apply U_γ conditioned on the ancilla qubit at the γth step of this construction.

A naive way of obtaining the Boolean products is to implement them via multiple-controlled Toffoli gates with appropriate control negations, which has total gate complexity $\mathcal{O}(\Gamma \log \Gamma)$. We give an improved implementation with gate complexity $\mathcal{O}(\Gamma)$ based on walking on a binary tree, which meets a previously established lower bound [80, Lemma 4]. A complete discussion of this improvement is beyond the scope of this dissertation, and we refer the reader to [34, Appendix G.4] for details.

3.4 Quantum-signal-processing implementation details

We now consider the quantum-signal-processing algorithm of Low and Chuang [73, 74], as introduced in Section 2.6. We describe optimizations that reduce the gate count of implementing quantum signal processing. We then discuss the difficulty of computing the phases that specify the algorithm and describe a segmented version of the algorithm that mitigates this issue. Finally, we describe empirical bounds on the error in the truncated Jacobi-Anger expansion and in the overall

algorithm.

Circuit optimizations. The select(H) gate is a major component of the quantum-signal-processing algorithm, so we use the optimized implementation of that subroutine [34, Appendix G.4] in our implementation of the quantum-signal-processing algorithm. We now present some further circuit optimizations that also reduce the gate count.

As discussed in Section 2.6, we use the phased iterate V_ϕ defined in (2.45), whereas Low and Chuang use the operation

$$V'_\phi := (e^{-i\phi Z/2} \otimes I)\Big(|+\rangle\langle+| \otimes I + |-\rangle\langle-| \otimes \big(Z_{-\pi/2}(-iQ)Z_{\pi/2}\big)\Big)(e^{i\phi Z/2} \otimes I), \quad (3.87)$$

where $Z_\varphi := (1 + e^{-i\varphi})|G\rangle\langle G| - I$ is a partial reflection about $|G\rangle$. It is easy to see that

$$V_\phi = (I \otimes Z_{\pi/2})V'_\phi(I \otimes Z_{-\pi/2}), \quad (3.88)$$

so

$$V_\phi^\dagger = (I \otimes Z_{\pi/2})V_\phi'^\dagger(I \otimes Z_{-\pi/2}) \quad (3.89)$$

also involves conjugation by $I \otimes Z_{\pi/2}$. Thus, when the phased iterates are applied in the sequence (2.49), the inner partial reflection gates cancel. Furthermore, Z_φ

simply introduces a relative phase between $|G\rangle$ and its orthogonal subspace, so its action is trivial if the ancilla register is initialized in and postselected on $|G\rangle$. Thus we see that the implementation as defined in Section 2.6 has the same effect as in [74]. Each partial reflection is implemented using $\mathcal{O}(\log n)$ elementary gates, and there are $\mathcal{O}(n^2)$ phased iterates, so our implementation saves $\mathcal{O}(n^2 \log n)$ gates.

We apply a similar simplification to further reduce the gate count. For every phased iterate V_ϕ defined in (2.45), we must implement a controlled version of the operator $-iQ = -i\big((2|G\rangle\langle G| - I) \otimes I\big)\,\mathrm{select}(H)$. In particular, this requires us to perform a controlled-reflection about $|G\rangle$. We can do this by performing a controlled-$U^\dagger$ that unprepares the state $|G\rangle$ (where U is a unitary operation satisfying $U|0\rangle = |G\rangle$), a controlled reflection about $|0\rangle$, and finally a controlled-U that prepares the state $|G\rangle$. However, observe that we can replace the controlled unitary conjugation by its uncontrolled version without changing the behavior of the circuit. Furthermore, by grouping neighboring pairs of phased iterates in the sequence of V_ϕ and $V_{\phi'}^\dagger$ operations, we can cancel pairs of unitary operators U and $U^\dagger$ for state preparation and unpreparation.

Phase computation and segmented algorithm. Recall that to specify the quantum-signal-processing algorithm, we must find phases $\phi_1, \ldots, \phi_m$ that realize the truncated Jacobi-Anger expansion. In principle, these angles can be computed in polynomial time [77]. However, this computation is difficult in practice, so we can only carry it out for very small instances. Specifically, we found the time

required to calculate the angles to be prohibitive for values of m greater than about 32. For $n = 10$ qubits with $t = n$ and $\epsilon = 10^{-3}$, the error bound (2.52) suggests that we should take $m = 1100$. Thus the difficulty of computing the angles prevents us from synthesizing nontrivial instances of the algorithm. This difficulty arises because the procedure for computing the angles requires us to compute the roots of a high-degree polynomial to high precision. It is a natural open problem to give a more practical method for computing the angles.

Fortunately, to determine the Clifford+R_z gate count in our implementation of the quantum-signal-processing algorithm, we do not need to know the angles of the phased iterates. Furthermore, since most R_z gates require approximately the same number of Hadamard and T gates to realize within a given precision, we can get a reasonable estimate of the Clifford+T count by using random angles in place of the true values. However, we emphasize that this method does not produce a correct quantum simulation circuit, and should only be used as a benchmark of the resource requirements of the quantum-signal-processing algorithm—which is only useful if the true angles can ultimately be computed.

An alternative is to consider what we call a *segmented* version of the algorithm. In this approach, we first divide the evolution time into r segments, each of which is sufficiently short that the angles can readily be computed. Since the optimality of the quantum-signal-processing approach to Hamiltonian simulation relies essentially on simulating the entire evolution as a single segment, the seg-

mented approach has higher asymptotic complexity. However, it allows us to give a complete implementation, and the overhead for moderate values of n is not too great.

To analyze the algorithm with r segments, we apply the error bound (2.52) with t replaced by t/r and ϵ replaced by ϵ/r. This gives the sufficient condition

$$\frac{4(\|\alpha\|_1 \, t/r)^q}{2^q \, q!} \leq \frac{\epsilon}{8r} \tag{3.90}$$

where $q = \frac{m}{2} + 1$. Thus

$$r = \mathcal{O}\left(\alpha t \left(\frac{\|\alpha\|_1 t}{2\epsilon(\frac{M}{2}+1)!}\right)^{2/m}\right) \tag{3.91}$$

segments suffice to ensure overall error at most ϵ. With $t = n$, $\alpha = \mathcal{O}(n)$, and m a fixed constant, we have $r = \mathcal{O}(n^{2+4/m})$ segments. Within each segment, the number of phased iterates is m, which is independent of the system size. Each phased iterate has circuit size $\mathcal{O}(n)$ using the improved select(V) implementation [34, Appendix G.4]. Therefore, the segmented algorithm has gate complexity $\mathcal{O}(n^{3+4/m})$.

In our implementation, we use $m = 28$ (i.e., $q = 15$). For the instance of quantum simulation considered in this paper, we set $\epsilon = 10^{-3}$, $\|\alpha\|_1 = 4n$, and

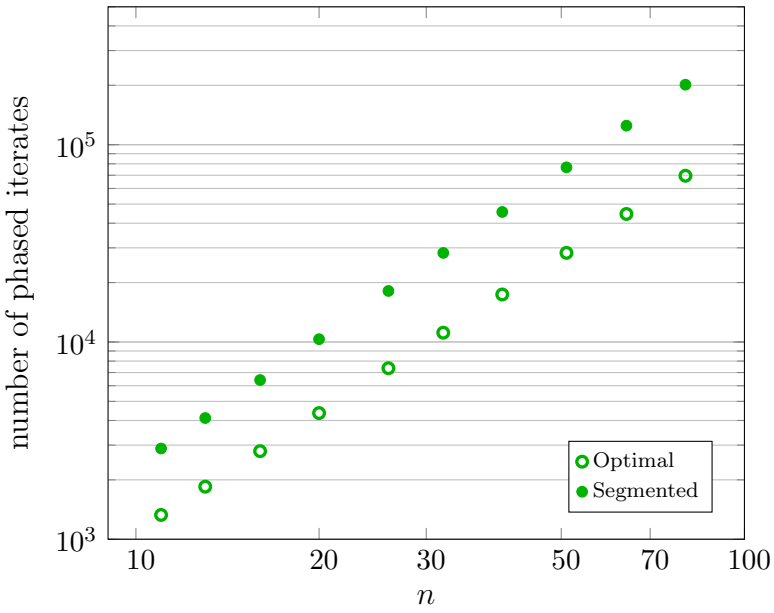

Figure 3.3: Comparison of the number of phased iterates using optimal and segmented implementations of the quantum-signal-processing algorithm.

$t = n$. With these values, (3.90) shows that it suffices to use

$$r \geq \sqrt[14]{\frac{10^3 \cdot 2^{20} n^{30}}{15!}} = 0.6010\, n^{15/7} \tag{3.92}$$

segments.

Figure 3.3 compares the total number of phased iterates used in the segmented and optimal implementations. Over the range of interest, the segmented algorithm is only worse by a factor between 2 and 3.

Empirical error bounds. The error bound (2.52) uses the closed-form expression (2.51) for the remainder of the Jacobi-Anger expansion. While it is a convenient to use such an analytical expression, it is natural to ask how tightly it bounds the complexity of the algorithm.

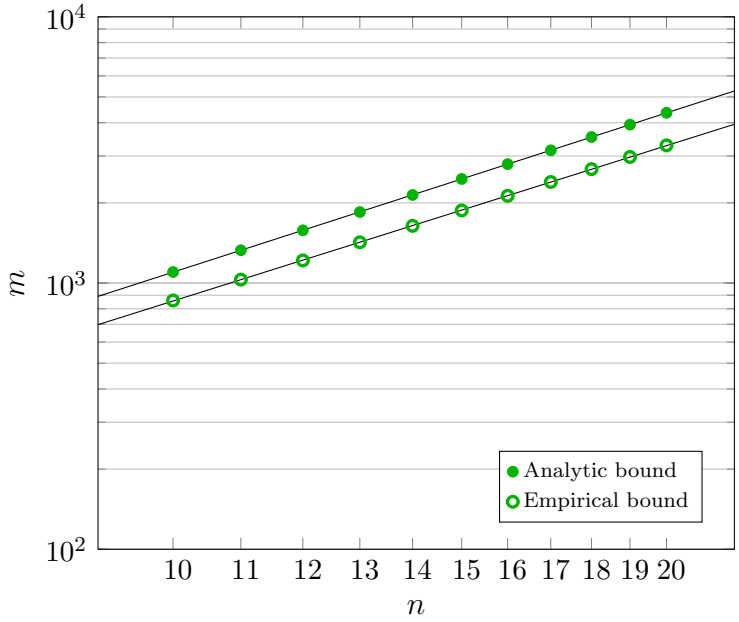

Figure 3.4: Comparison of the number of phased iterates using the analytic bound (2.51) and the empirical bound for the Jacobi-Anger expansion. Here m is the number of phased iterates and n is the system size.

To address this question, we numerically evaluate the left-hand side of (2.51) for systems of sizes ranging from 10 to 20, as shown in Figure 3.4. By extrapolating these data, we estimate the complexity of the quantum-signal-processing algorithm for arbitrary sizes, including those for which classical evaluation of the series is intractable. The empirical bound improves the gate count by a factor between 1.25 and 1.45 over the range of interest ($10 \leq n \leq 100$). More specifically, power law fits to the data give

$$m_{\text{ana}} = 11.30\, n^{1.988}, \quad m_{\text{emp}} = 9.849\, n^{1.939} \tag{3.93}$$

for the number of phased iterates using either the analytic bound or the empirical

bound, respectively. Since each phased iterate has gate complexity $\mathcal{O}(n)$ (using the technique from [34, Appendix G.4]), we find that the quantum-signal-processing algorithm has complexity $\mathcal{O}(n^{2.988})$ (resp., $\mathcal{O}(n^{2.939})$) using the analytic bound (resp., empirical bound).

We do not consider the empirical bound for the segmented version of the quantum-signal-processing algorithm, since the savings is small in that case (even less at $m = 28$ than at the values shown in Figure 3.4), and the main goal of the segmented approach is to have a fully-specified algorithm with rigorous guarantees. However, we use the empirical bound to estimate resources using the non-segmented quantum-signal-processing algorithm. This produces our most optimistic benchmark for the performance of the quantum-signal-processing algorithm.

One could also consider a full empirical estimate for the quantum-signal-processing algorithm by using direct simulation to determine its true overall error. The need for ancilla qubits makes this challenging: the algorithm uses $n+\lceil \log 4n \rceil + 1$ qubits to simulate an n-qubit system. Fortunately, unlike with the Taylor-series algorithm, small instances of the quantum-signal-processing algorithm are just within reach of direct classical simulation.

However, preliminary numerical investigation suggests that the performance of the quantum-signal-processing algorithm cannot be significantly improved using such an empirical bound. Figure 3.5 shows the empirical error in the segmented

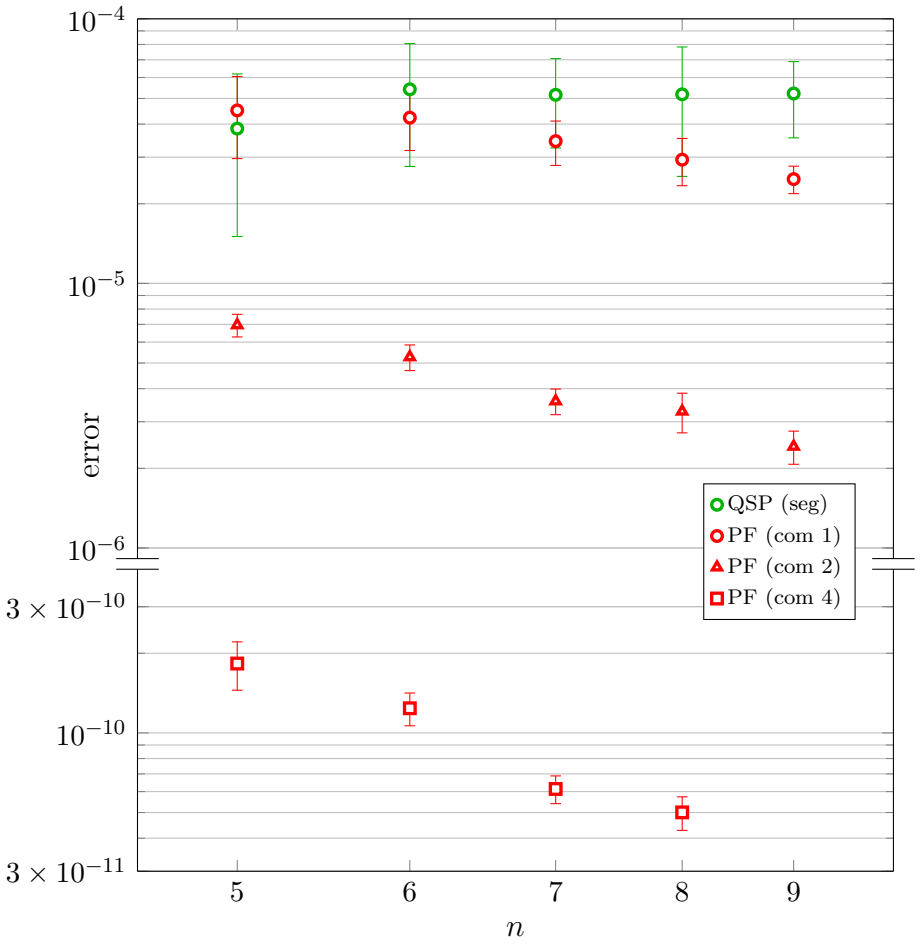

Figure 3.5: Empirical error in the segmented quantum-signal-processing algorithm and product-formula algorithms of orders 1, 2, and 4 (with commutator bound) for small system sizes.

quantum-signal-processing algorithm for small system sizes, averaging over 10 random experiments with a fixed target error $\epsilon = 10^{-3}$, along with similar data for the product-formula algorithm using the commutator bound. We observe that for system sizes between 5 and 9, the quantum-signal-processing error is consistently around 5×10^{-5}, which is not significantly less than the target error of 10^{-3}. While there was more variation in the error of the quantum-signal-processing algorithm as compared to the product-formula algorithm, in no case was the quantum-signal-processing error less than 10^{-5}. In contrast to the product-formula algorithm, where the error apparently decreases as a power law in n, the quantum-signal-processing error shows no indication of decreasing. Furthermore, since the complexity of the quantum-signal-processing algorithm depends logarithmically on the inverse error $1/\epsilon$, even a large reduction in the error may not have a significant effect. For these reasons, we do not consider full empirical error bounds in our resource estimates for the quantum-signal-processing algorithm.

3.5 Results

We implement the three simulation algorithms in a quantum circuit description language called Quipper [51]. We also process all circuits using an automated tool we developed for large-scale quantum circuit optimization [82]. Our implementation is available in a public repository [33].

We express our circuits over the set of two-qubit CNOT gates, single-qubit

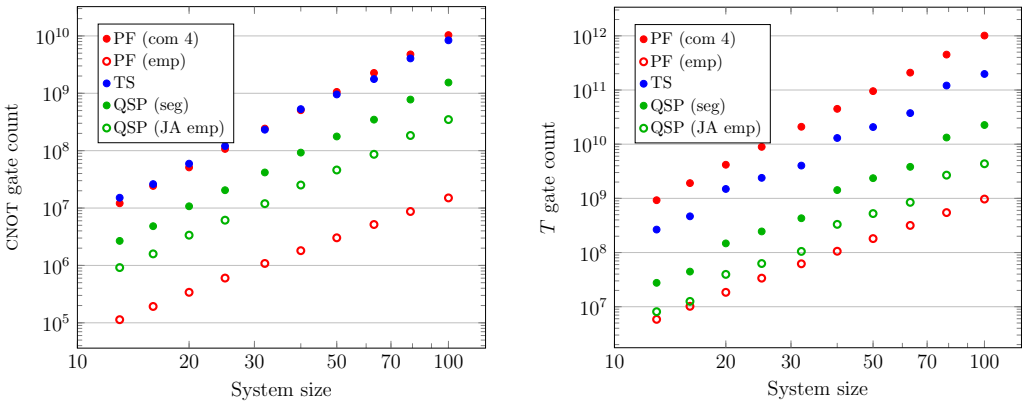

Figure 3.6: Gate counts for optimized implementations of the product-formula (PF) algorithm (using the fourth-order formula with commutator bound and the better of the fourth- or sixth-order formula with empirical error bound), the Taylor-series (TS) algorithm, and the quantum-signal-processing (QSP) algorithm (using the segmented version with analytic error bound and the non-segmented version with empirical Jacobi-Anger error bound) for system sizes between 10 and 100. Left: CNOT gates for Clifford+R_z circuits. Right: T gates for Clifford+T circuits.

Clifford gates, and single-qubit Z rotations $R_z(\theta) := \exp(-iZ\theta/2)$ for $\theta \in \mathbb{R}$, which can be directly implemented with both trapped ions [40] and superconducting circuits [28, 62]. We focus on the CNOT count as two-qubit gates take longer to perform and incur more error than single-qubit gates. We also produce Clifford+T circuits using optimal circuit synthesis [92] so that we can count T gates, which are typically the most expensive gates for fault-tolerant computation.

Figure 3.6 and Figure 3.7 compare the gate counts and qubit counts for the product-formula (PF) algorithm (with commutator and empirical error bounds), the Taylor-series (TS) algorithm, and the quantum-signal-processing (QSP) algorithm (in both segmented and non-segmented versions). Among all the algorithms we considered, the PF algorithm does not need ancilla qubits, making it suitable for near-term implementation of quantum simulation. The implementation of

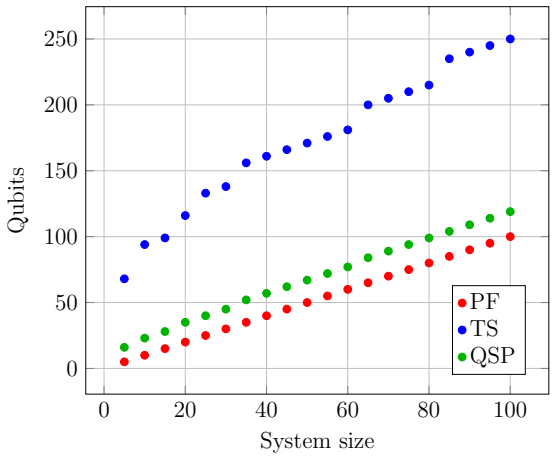

Figure 3.7: Number of qubits used by the product-formula (PF), Taylor-series (TS), and quantum-signal-processing (QSP) algorithms.

other algorithms uses ancilla qubits, but QSP has mild space requirement and is preferred over the TS approach.

Despite being more involved, the segmented QSP algorithm has the best performance among the rigorously-analyzed algorithms. However, the performance of the PF algorithm is significantly improved using the empirical bounds, making it the preferred approach if rigorous performance guarantees are not required, especially considering its lower space requirement. This significant gap between the provable and actual performance of product formulas may be closed by proving stronger error bounds, which we further discuss in Chapter 6 and Chapter 7.

Although higher-order product formulas have been deemphasized in recent work of quantum chemistry simulation [87, 90], we find that they are surprisingly efficient for simulating systems of small sizes, as shown in Figure 3.8. The fourth-order formula with commutator bound gives the best available PF algorithm with

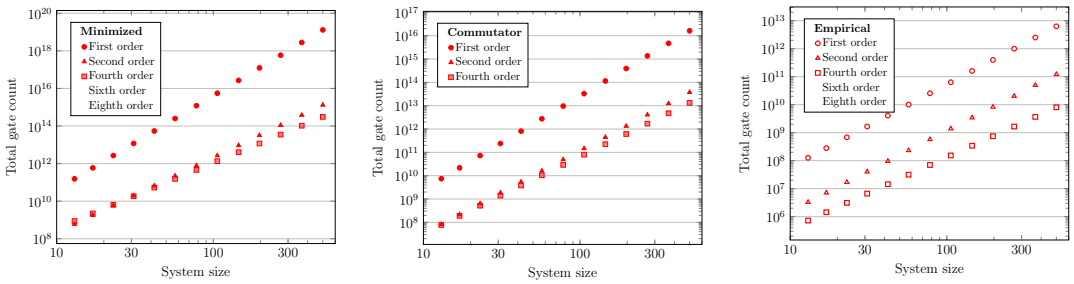

Figure 3.8: Total gate counts in the Clifford+R_z basis for product formula algorithms using the minimized (left), commutator (center), and empirical (right) bounds, for system sizes between 13 and 500.

a rigorous performance guarantee, whereas the sixth-order formula outperforms the fourth-order formula for systems of about 30 or more qubits using empirical error bounds. For future work, it could be fruitful to experimentally demonstrate the utility of these higher-order formulas.

For a system of 50 qubits—which is presumably close to the limits of direct classical simulation for circuits such as ours—the segmented QSP algorithm is the best rigorously-analyzed approach, using about 1.8×10^8 CNOT gates (over the set of Clifford+R_z gates) and 2.4×10^9 T gates (over the set of Clifford+T gates). This is further reduced using the PF algorithm with empirical bounds, costing about 3×10^6 CNOTs and 1.8×10^8 Ts (over Clifford+R_z and Clifford+T, respectively). For comparison, previous estimates of gate counts for factoring, discrete logarithms, and quantum chemistry simulations are significantly larger (Figure 3.9). This suggests that simulation of spin systems is a significantly easier task for near-term practical quantum computation.

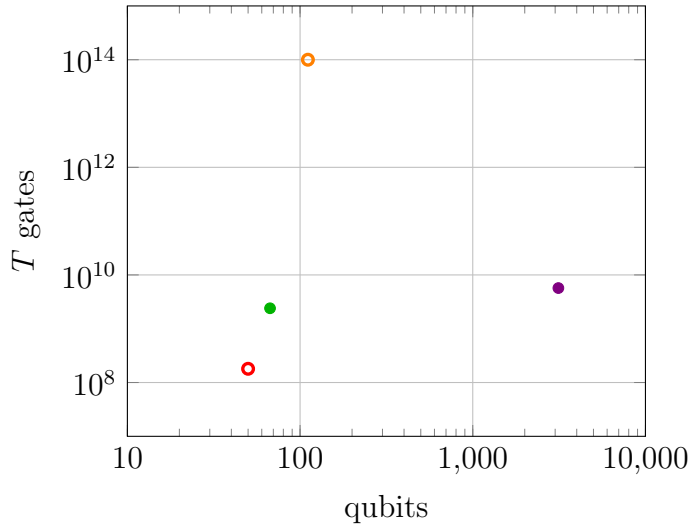

Figure 3.9: Comparison of resource requirements for solving classically infeasible, practically useful problem instances on a quantum computer: factoring a 1024-bit number [69] (purple), simulation of FeMoco [90] (orange), and 50-spin simulations described in this paper (segmented QSP in green; sixth-order PF with empirical error bound in red).

Chapter 4: Randomized product formulas

In this chapter, we propose a randomized approach to quantum simulation based on higher-order product formulas. The resulting algorithm is not much more complicated than a deterministic product formula, but the savings in the simulation cost can be substantial.

Our analysis uses a mixing lemma of Campbell and Hastings [24, 56] to bound the diamond norm distance of the actual operator from the ideal evolution. We motivate this approach in Section 4.1, where we consider the effect of randomizing how the summands are ordered in the simple case of the first-order formula.

Analyzing the effect of randomization on higher-order formulas is more challenging. For low-order terms in the Taylor expansion of a product formula, the majority of the error comes from terms in which no summands are repeated. We call such contributions *nondegenerate terms*. In Section 4.2, we give a combinatorial argument to show that the nondegenerate terms completely cancel in the randomized product formula.

Section 4.3 presents our main technical result, an upper bound on the error in a randomized higher-order product-formula simulation. This bound follows by applying the mixing lemma to combine an error bound for the average evolution operator with standard product formula error bounds for the error of the individual terms. Section 4.4 discusses the overall performance of the resulting algorithm

and compares it with deterministic approaches. Our algorithm always improves the dependence on the number of Hamiltonian summands and sometimes achieves better dependence on the evolution time and simulation accuracy as well.

We also show in Section 4.4 that our bound can outperform a previous bound that takes advantage of the structure of the Hamiltonian. Specifically, we compare our randomized product formula algorithm with the deterministic algorithm using the commutator bound of [34] for a one-dimensional Heisenberg model in a random magnetic field. We find that over a significant range of parameters, the randomized algorithm has better proven performance, despite using less information about the form of the Hamiltonian.

In light of the large gap between proven and empirical performance of product formulas, it is natural to ask whether randomized product formulas still offer an improvement under the best possible error bounds. To address this question, we present numerical comparisons of the deterministic and randomized product formulas in Section 4.5. In particular, we show that the randomized approach can sometimes outperform the deterministic approach even with respect to their empirical performance. Finally, we conclude in Section 4.6 with a brief discussion of the results and some open questions.

This chapter is partly based on the following paper:

[35] Andrew M. Childs, Aaron Ostrander, and Yuan Su, *Faster quantum simulation by randomization*, Quantum **3** (2019), 182, arXiv:1805.08385.

4.1 The power of randomization

To see how randomness can improve a product formula simulation, consider a simple Hamiltonian expressed as a sum of two operators, $H = H_1 + H_2$. The Taylor expansion of the first-order formula as a function of $\lambda \in \mathbb{C}$ is

$$\mathscr{S}_1(\lambda) = \exp(\lambda H_1)\exp(\lambda H_2) = I + \lambda(H_1 + H_2) + \frac{\lambda^2}{2}(H_1^2 + 2H_1H_2 + H_2^2) + O(\lambda^3), \tag{4.1}$$

whereas the Taylor series of the ideal evolution is

$$\mathscr{V}(\lambda) = \exp((H_1+H_2)\lambda) = I+\lambda(H_1+H_2)+\frac{\lambda^2}{2}(H_1^2+H_1H_2+H_2H_1+H_2^2)+O(\lambda^3). \tag{4.2}$$

Using the triangle inequality, we can bound the spectral-norm error as

$$\|\mathscr{V}(\lambda) - \mathscr{S}_1(\lambda)\| \leq \|[H_1, H_2]\| \frac{|\lambda|^2}{2} + \mathcal{O}((\Lambda|\lambda|)^3), \tag{4.3}$$

where $\Lambda := \max\{\|H_1\|, \|H_2\|\}$. Since H_1 and H_2 need not commute, $\mathscr{S}_1(\lambda)$ approximates $\mathscr{V}(\lambda)$ to first order in λ, as expected.

It is clearly impossible to approximate $\mathscr{V}(\lambda)$ to second order using a product of only two exponentials of H_1 and H_2: any such product can have only one of the products H_1H_2 and H_2H_1 in its Taylor expansion, whereas $\mathscr{V}(\lambda)$ contains both of these products in its second-order term. However, we can obtain both products by

taking a uniform mixture of $\mathscr{S}_1(\lambda)$ and

$$\mathscr{S}_1^{\mathrm{rev}}(\lambda) := \exp(\lambda H_2)\exp(\lambda H_1). \tag{4.4}$$

Indeed, a simple calculation shows that

$$\left\| \mathscr{V}(\lambda) - \frac{1}{2}\big(\mathscr{S}_1(\lambda) + \mathscr{S}_1^{\mathrm{rev}}(\lambda)\big) \right\| = \mathcal{O}\big((\Lambda|\lambda|)^3\big). \tag{4.5}$$

However, $\big(\mathscr{S}_1(-it)+\mathscr{S}_1^{\mathrm{rev}}(-it)\big)/2$ is not a unitary operation in general. We could in principle implement a linear combination of unitaries using the techniques of [13], but such an approach would use ancillas and could have high cost, especially when the Hamiltonian contains many summands. A simpler approach is to apply one of the two operations $\mathscr{S}_1(-it)$ and $\mathscr{S}_1^{\mathrm{rev}}(-it)$ chosen uniformly at random (as in Algorithm 2 of [113]), thereby implementing a quantum channel that gives a good approximation to the desired evolution.

We now introduce some notation that is useful to analyze the performance of randomized product formulas. Let X be a matrix acting on a finite-dimensional Hilbert space $\mathcal{H}$. We write $\|X\|$ for its spectral norm (the largest singular value) and $\|X\|_{\mathrm{tr}}$ for its trace norm (the sum of its singular values, i.e., its Schatten 1-norm). Let $\mathcal{E}\colon X \mapsto \mathcal{E}(X)$ be a linear map on the space of matrices on $\mathcal{H}$. The

diamond norm of $\mathcal{E}$ is [106, 112]

$$\|\mathcal{E}\|_\diamond := \max\{\|(\mathcal{E}\otimes \mathbb{1}_{\mathcal{H}})(Y)\|_{\mathrm{tr}} : \|Y\|_{\mathrm{tr}} \leq 1\}, \tag{4.6}$$

where the maximization is taken over all matrices Y on $\mathcal{H}\otimes\mathcal{H}$ satisfying $\|Y\|_{\mathrm{tr}} \leq 1$.

The following mixing lemma bounds how well we can approximate a unitary operation using a random unitary channel. Specifically, the error is linear in the distance between the target unitary and the average of the random unitaries, and only quadratic in the distance between the target unitary and each individual random unitary.

Lemma 21 (Mixing lemma [24, 56])**.** *Let V and $\{U_j\}$ be unitary matrices, with associated quantum channels $\mathcal{V}\colon \rho \mapsto V\rho V^\dagger$ and $\mathcal{U}_j\colon \rho \mapsto U_j\rho U_j^\dagger$, and let $\{p_j\}$ be a collection of positive numbers satisfying $\sum_j p_j = 1$. Suppose that*

(i) $\|U_j - V\| \leq a$ *for all* j *and*

(ii) $\left\|(\sum_j p_j U_j) - V\right\| \leq b$.

Then the average evolution $\mathcal{E} := \sum_j p_j \mathcal{U}_j$ satisfies $\|\mathcal{E} - \mathcal{V}\|_\diamond \leq a^2 + 2b$.

To simulate the Hamiltonian $H = H_1 + H_2$ for time t, we divide the evolution into r segments of duration t/r and implement each segment via the random unitary operation

$$\frac{1}{2}\big(\mathcal{S}_1(-it/r) + \mathcal{S}_1^{\mathrm{rev}}(-it/r)\big) \tag{4.7}$$

using one bit of randomness per segment, where $\mathcal{S}_1$ and $\mathcal{S}_1^{\text{rev}}$ are the quantum channels associated with $\mathscr{S}_1$ and $\mathscr{S}_1^{\text{rev}}$. Invoking the mixing lemma with $a = \mathcal{O}\big((\Lambda t)^2/r^2\big)$ and $b = \mathcal{O}\big((\Lambda t)^3/r^3\big)$, we find that

$$\left\| \mathcal{V}(-it/r) - \frac{1}{2}\big(\mathcal{S}_1(-it/r) + \mathcal{S}_1^{\text{rev}}(-it/r)\big) \right\|_\diamond = \mathcal{O}\left(\frac{(\Lambda t)^3}{r^3}\right). \tag{4.8}$$

Since the diamond norm distance between quantum channels is subadditive under composition [106, p. 178], the error of the entire simulation is

$$\left\| \mathcal{V}(-it) - \frac{1}{2^r}\big(\mathcal{S}_1(-it/r) + \mathcal{S}_1^{\text{rev}}(-it/r)\big)^r \right\|_\diamond = \mathcal{O}\left(\frac{(\Lambda t)^3}{r^2}\right). \tag{4.9}$$

Thus the randomized first-order formula is effectively a second-order formula.

This approach easily extends to a sum of Γ operators, again effectively making the first-order formula accurate to second order (cf. [113], which shows the same result with respect to trace distance of the output state). Keeping track of all the prefactors, we find the following error bound for the randomized first-order formula.

Theorem 22 (Randomized first-order error bound)**.** *Let* $\{H_\gamma\}_{\gamma=1}^\Gamma$ *be Hermitian matrices. Let*

$$\mathscr{V}(-it) := \exp\left(-it \sum_{\gamma=1}^{\Gamma} H_\gamma\right) \tag{4.10}$$

be the evolution induced by the Hamiltonian $H = \sum_{\gamma=1}^\Gamma H_\gamma$ *for time* $t \in \mathbb{R}$*. Define*

$$\mathscr{S}_1(\lambda) := \prod_{\gamma=1}^{\Gamma} \exp(\lambda H_\gamma) \quad \textit{and} \quad \mathscr{S}_1^{\mathrm{rev}}(\lambda) := \prod_{\gamma=\Gamma}^{1} \exp(\lambda H_\gamma). \tag{4.11}$$

Let $r \in \mathbb{N}$ be a positive integer and $\Lambda := \max_\gamma \|H_\gamma\|$. Then

$$\begin{aligned}\left\| \mathcal{V}(-it) - \frac{1}{2^r}\left(\mathcal{S}_1(-it/r) + \mathcal{S}_1^{\mathrm{rev}}(-it/r)\right)^r \right\|_\diamond \leq & \frac{(\Lambda|t|\Gamma)^4}{r^3} \exp\left(2\frac{\Lambda|t|\Gamma}{r}\right) \\ & + \frac{2(\Lambda|t|\Gamma)^3}{3r^2} \exp\left(\frac{\Lambda|t|\Gamma}{r}\right)\end{aligned} \tag{4.12}$$

where, for $\lambda = -it$, we associate channels $\mathcal{V}(\lambda)$, $\mathcal{S}_1(\lambda)$, and $\mathcal{S}_1^{\mathrm{rev}}(\lambda)$ with the unitaries $\mathscr{V}(\lambda)$, $\mathscr{S}_1(\lambda)$, and $\mathscr{S}_1^{\mathrm{rev}}(\lambda)$, respectively.

To guarantee that the simulation error is at most ϵ, we upper bound the right-hand side of (4.12) by ϵ and solve for r. Assuming $\Lambda := \max_\gamma \|H_\gamma\|$ is constant, we find that it suffices to choose $r_1^{\mathrm{rand}} = \mathcal{O}\big((t\Gamma)^{1.5}/\epsilon^{0.5}\big)$, giving a simulation algorithm with gate complexity $g_1^{\mathrm{rand}} = \mathcal{O}\big(t^{1.5}\Gamma^{2.5}/\epsilon^{0.5}\big)$. In comparison, the gate complexity in the deterministic case is $g_1^{\mathrm{det}} = \mathcal{O}\big(t^2\Gamma^3/\epsilon\big)$. Therefore, the randomized first-order product formula algorithm improves over the deterministic algorithm with respect to all parameters of interest.

It is natural to ask whether a similar randomization strategy can improve higher-order product formulas (as defined in (2.14)). While it turns out that randomization does not improve the order of the formula, it does result in a significant reduction of the error, and in particular, lowers the dependence on the number of summands in the Hamiltonian. The more complicated structure of higher-order

formulas makes this analysis more involved than in the first-order case (in particular, we randomly permute the Γ summands instead of simply choosing whether or not to reverse them, so we use $\Theta(\Gamma \log \Gamma)$ bits of randomness per segment instead of only a single bit). As aforementioned, our proof is based on a randomization lemma (established in the next section) that evaluates the dominant contribution to the Taylor series of the randomized product formula in closed form.

4.2 Randomization lemma

In this section, we study the Taylor expansion of the average evolution operator obtained by randomizing how the summands of a Hamiltonian are ordered. We consider a formula of the form

$$\begin{aligned}
&\exp(q_1 \lambda H_{\pi_1(1)}) \exp(q_1 \lambda H_{\pi_1(2)}) \cdots \exp(q_1 \lambda H_{\pi_1(\Gamma)}) \\
&\exp(q_2 \lambda H_{\pi_2(1)}) \exp(q_2 \lambda H_{\pi_2(2)}) \cdots \exp(q_2 \lambda H_{\pi_2(\Gamma)}) \\
&\cdots \\
&\exp(q_\kappa \lambda H_{\pi_\kappa(1)}) \exp(q_\kappa \lambda H_{\pi_\kappa(2)}) \cdots \exp(q_\kappa \lambda H_{\pi_\kappa(\Gamma)})
\end{aligned} \tag{4.13}$$

for real numbers $q_1, \ldots, q_\kappa \in \mathbb{R}$, a complex number $\lambda \in \mathbb{C}$, Hermitian matrices $H_1, \ldots, H_\Gamma$, and permutations $\pi_1, \ldots, \pi_\kappa \in \mathrm{Sym}(\Gamma)$. By choosing appropriate values of $q_1, \ldots, q_\kappa \in \mathbb{R}$ and ordering $H_1, \ldots, H_\Gamma$ in both forward and backward directions, we can write any product formula $\mathscr{S}_{2k}(\lambda)$ in this form.

We now permute the summands to get the average evolution

$$\begin{aligned}\frac{1}{\Gamma!}\sum_{\sigma\in\mathrm{Sym}(\Gamma)} &\exp(q_1\lambda H_{\sigma(\pi_1(1))})\exp(q_1\lambda H_{\sigma(\pi_1(2))})\cdots\exp(q_1\lambda H_{\sigma(\pi_1(\Gamma))})\\ &\exp(q_2\lambda H_{\sigma(\pi_2(1))})\exp(q_2\lambda H_{\sigma(\pi_2(2))})\cdots\exp(q_2\lambda H_{\sigma(\pi_2(\Gamma))})\\ &\cdots\\ &\exp(q_\kappa\lambda H_{\sigma(\pi_\kappa(1))})\exp(q_\kappa\lambda H_{\sigma(\pi_\kappa(2))})\cdots\exp(q_\kappa\lambda H_{\sigma(\pi_\kappa(\Gamma))}).\end{aligned} \tag{4.14}$$

In its Taylor expansion, we call the sum of the form

$$\sum_{\substack{m_1,\ldots,m_s\\ \text{pairwise different}}} \alpha_{m_1\ldots m_s}\lambda^s H_{m_1}\cdots H_{m_s}, \tag{4.15}$$

with coefficients $\alpha_{m_1\ldots m_s}\in\mathbb{C}$, the sth-order *nondegenerate term*. This term contributes $\Theta(\Gamma^s)$ to the sth-order error, whereas the remaining (degenerate) terms only contribute $\mathcal{O}(\Gamma^{s-1})$.

The following lemma shows how to compute the sth-order nondegenerate term for an arbitrary average evolution.

Lemma 23 (Randomization lemma)**.** *Define an average evolution operator as in (4.14) and let $s\le\Gamma$ be a positive integer. The sth-order nondegenerate term of this operator is*

$$\frac{[(q_1+\cdots+q_\kappa)\lambda]^s}{s!}\sum_{\substack{m_1,\ldots,m_s\\ \textit{pairwise different}}} H_{m_1}\cdots H_{m_s}. \tag{4.16}$$

Proof. We take all possible products of s terms from the Taylor expansion of (4.14). Observe that the exponentials in (4.14) are organized in an array with κ rows and Γ columns. We use $\kappa_1, \ldots, \kappa_s$ and $l_1, \ldots, l_s$ to label the row and column indices, respectively, of the exponentials from which the terms are chosen. To avoid double counting, we take terms with smaller row indices first (i.e., $\kappa_1 \leq \cdots \leq \kappa_s$). Within each row, we take terms with smaller column indices first. To get the sth-order nondegenerate term, we require that $\pi_{\kappa_1}(l_1), \ldots, \pi_{\kappa_s}(l_s)$ are pairwise different. The sth-order nondegenerate term of (4.14) can then be expressed as

$$\frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \sum_{\kappa_1 \leq \cdots \leq \kappa_s} \sum_{\substack{\pi_{\kappa_1}(l_1), \ldots, \pi_{\kappa_s}(l_s) \\ \text{pairwise different}}} (q_{\kappa_1} \lambda H_{\sigma(\pi_{\kappa_1}(l_1))}) \cdots (q_{\kappa_s} \lambda H_{\sigma(\pi_{\kappa_s}(l_s))}). \tag{4.17}$$

A direct calculation shows that

$$\begin{aligned}
&\frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \sum_{\kappa_1 \leq \cdots \leq \kappa_s} \sum_{\substack{\pi_{\kappa_1}(l_1), \ldots, \pi_{\kappa_s}(l_s) \\ \text{pairwise different}}} (q_{\kappa_1} \lambda H_{\sigma(\pi_{\kappa_1}(l_1))}) \cdots (q_{\kappa_s} \lambda H_{\sigma(\pi_{\kappa_s}(l_s))}) \\
&= \frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \sum_{\kappa_1 \leq \cdots \leq \kappa_s} \sum_{\substack{\pi_{\kappa_1}(l_1), \ldots, \pi_{\kappa_s}(l_s) \\ \text{pairwise different}}} \sum_{\substack{m_1 = \sigma(\pi_{\kappa_1}(l_1)), \ldots, \\ m_s = \sigma(\pi_{\kappa_s}(l_s))}} (q_{\kappa_1} \lambda H_{m_1}) \cdots (q_{\kappa_s} \lambda H_{m_s}) \\
&= \frac{1}{\Gamma!} \sum_{\substack{m_1, \ldots, m_s \\ \text{pairwise different}}} \sum_{\kappa_1 \leq \cdots \leq \kappa_s} \sum_{\substack{\pi_{\kappa_1}(l_1), \ldots, \pi_{\kappa_s}(l_s) \\ \text{pairwise different}}} \sum_{\substack{\sigma \in \mathrm{Sym}(\Gamma): \\ \sigma(\pi_{\kappa_1}(l_1)) = m_1, \ldots, \\ \sigma(\pi_{\kappa_s}(l_s)) = m_s}} (q_{\kappa_1} \lambda H_{m_1}) \cdots (q_{\kappa_s} \lambda H_{m_s}) \\
&= \frac{(\Gamma - s)!}{\Gamma!} \sum_{\substack{m_1, \ldots, m_s \\ \text{pairwise different}}} \left[\sum_{\kappa_1 \leq \cdots \leq \kappa_s} \sum_{\substack{\pi_{\kappa_1}(l_1), \ldots, \pi_{\kappa_s}(l_s) \\ \text{pairwise different}}} (q_{\kappa_1} \lambda) \cdots (q_{\kappa_s} \lambda) \right] H_{m_1} \cdots H_{m_s}.
\end{aligned} \tag{4.18}$$

Now observe that the summand $(q_{\kappa_1} \lambda) \cdots (q_{\kappa_s} \lambda)$ depends only on the row indices.

Letting $r_1, \ldots, r_\kappa$ denote the number of terms picked from row $1, \ldots, \kappa$, respectively, we can re-express this summand as $(q_1\lambda)^{r_1} \cdots (q_\kappa\lambda)^{r_\kappa}$. We determine the coefficient of this term as follows. The number of ways of choosing $l_1, \ldots, l_s$ pairwise different is $\Gamma(\Gamma-1)\cdots(\Gamma-s+1)$. However, when we apply permutations $\pi_{\kappa_1}, \ldots, \pi_{\kappa_s}$, we may double count some terms. In particular, if $\kappa_i = \kappa_{i+1}$, we are to pick terms from the same row κ_i and we must have $l_i < l_{i+1}$. This implies that the ordering of $\pi_{\kappa_i}(l_i)$ and $\pi_{\kappa_{i+1}}(l_{i+1})$ is uniquely determined. Altogether, we see that we have overcounted by a factor of $(r_1!)\cdots(r_\kappa!)$. Therefore, we have

$$\begin{aligned}\sum_{\kappa_1 \le \cdots \le \kappa_s} \sum_{\substack{\pi_{\kappa_1}(l_1),\ldots,\pi_{\kappa_s}(l_s) \\ \text{pairwise different}}} (q_{\kappa_1}\lambda)\cdots(q_{\kappa_s}\lambda) &= \sum_{\substack{r_1,\ldots,r_\kappa: \\ r_1+\cdots+r_\kappa=s}} \frac{\Gamma(\Gamma-1)\cdots(\Gamma-s+1)}{(r_1!)\cdots(r_\kappa!)} (q_1\lambda)^{r_1}\cdots(q_\kappa\lambda)^{r_\kappa} \\ &= \Gamma(\Gamma-1)\cdots(\Gamma-s+1)\frac{[(q_1+\cdots+q_\kappa)\lambda]^s}{s!}, \end{aligned} \tag{4.19}$$

where the last equality follows by the multinomial theorem.

Substituting (4.19) into (4.18) completes the proof. □

As an immediate corollary, we compute the sth-order nondegenerate term of the average evolution operator $\frac{1}{\Gamma!}\sum_{\sigma\in\mathrm{Sym}(\Gamma)} \mathscr{S}_{2k}^{\sigma}(\lambda)$.

Corollary 24. *Let $\{H_\gamma\}_{\gamma=1}^{\Gamma}$ be Hermitian operators; let $\lambda \in \mathbb{C}$, $k, s \in \mathbb{N}$, and $s \le \Gamma$. Then the sth-order nondegenerate term of average evolution $\frac{1}{\Gamma!}\sum_{\sigma\in\mathrm{Sym}(\Gamma)} \mathscr{S}_{2k}^{\sigma}(\lambda)$,*

with $\mathscr{S}_{2k}^{\sigma}(\lambda)$ being the permuted $(2k)$th-order Suzuki formula

$$\begin{aligned}\mathscr{S}_{2}^{\sigma}(\lambda) &:= \prod_{\gamma=1}^{\Gamma} \exp\left(\frac{\lambda}{2} H_{\sigma(\gamma)}\right) \prod_{\gamma=\Gamma}^{1} \exp\left(\frac{\lambda}{2} H_{\sigma(\gamma)}\right) \\ \mathscr{S}_{2k}^{\sigma}(\lambda) &:= [\mathscr{S}_{2k-2}^{\sigma}(p_k \lambda)]^2 \mathscr{S}_{2k-2}^{\sigma}((1-4p_k)\lambda) [\mathscr{S}_{2k-2}^{\sigma}(p_k \lambda)]^2.\end{aligned} \tag{4.20}$$

is

$$\frac{\lambda^s}{s!} \sum_{\substack{m_1,\ldots,m_s \\ \textit{pairwise different}}} H_{m_1} \cdots H_{m_s}. \tag{4.21}$$

Proof. The fact that $\mathscr{S}_{2k}^{\sigma}(\lambda)$ is at least first-order accurate implies that $q_1 + \cdots + q_\kappa = 1$ in (4.16). □

Observe that the sth-order nondegenerate term of $\mathscr{V}(\lambda) = \exp\bigl(\lambda \sum_{\gamma=1}^{\Gamma} H_\gamma\bigr)$ is also given by (4.21). Therefore, the sth-order nondegenerate term completely cancels in

$$\mathscr{V}(\lambda) - \frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathscr{S}_{2k}^{\sigma}(\lambda). \tag{4.22}$$

4.3 Error bounds

In this section we establish our main result, an upper bound on the error of a randomized product formula simulation. To apply the mixing lemma, we need to bound the error of the average evolution. We now present an error bound for an arbitrary fixed-order term in the Taylor expansion of the average evolution operator.

Lemma 25. *Let $\{H_\gamma\}_{\gamma=1}^\Gamma$ be Hermitian operators; let $\lambda \in \mathbb{C}$ and $k, s \in \mathbb{N}$. Define the target evolution $\mathscr{V}(\lambda)$ as*

$$\mathscr{V}(\lambda) := \exp\left(\lambda \sum_{\gamma=1}^{\Gamma} H_\gamma\right) \tag{4.23}$$

and define the permuted $(2k)$th-order formula $\mathscr{S}_{2k}^\sigma(\lambda)$ as in (4.20). Then the sth-order error of the approximation

$$\mathscr{V}(\lambda) - \frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathscr{S}_{2k}^\sigma(\lambda) \tag{4.24}$$

is at most

$$\begin{cases} 0 & 0 \leq s \leq 2k, \\ \frac{(2 \cdot 5^{k-1} \Lambda |\lambda|)^s}{(s-2)!} \Gamma^{s-1} & s > 2k, \end{cases} \tag{4.25}$$

where $\Lambda := \max_\gamma \|H_\gamma\|$.

The proof of this error bound uses the following estimate of a fixed-order degenerate term in the average evolution operator.

Lemma 26. *Let $\{H_\gamma\}_{\gamma=1}^\Gamma$ be Hermitian operators with $\Lambda := \max_\gamma \|H_\gamma\|$; let $q_1, \ldots, q_\kappa \in \mathbb{R}$ with $\max_k |q_k| \leq 1$; and let $s \leq \Gamma$ be a positive integer. Then the norm of the sth-order degenerate term of the ideal evolution operator $\mathscr{V}(\lambda)$ as in (4.23) is at most*

$$\frac{(\Lambda|\lambda|)^s}{s!} \left[\Gamma^s - \Gamma(\Gamma-1)\cdots(\Gamma-s+1)\right] \tag{4.26}$$

and the norm of the sth-order degenerate term of the average evolution operator as in (4.14) is at most

$$\frac{(\kappa\Lambda|\lambda|)^s}{s!}\big[\Gamma^s - \Gamma(\Gamma-1)\cdots(\Gamma-s+1)\big]. \tag{4.27}$$

Proof. The sth-order term of $\mathscr{V}(\lambda)$ is

$$\frac{\big(\lambda\sum_{\gamma=1}^{\Gamma} H_\gamma\big)^s}{s!} = \frac{\lambda^s}{s!}\sum_{m_1,\ldots,m_s} H_{m_1}\cdots H_{m_s} \tag{4.28}$$

and its nondegenerate term is

$$\frac{\lambda^s}{s!}\sum_{\substack{m_1,\ldots,m_s\\ \text{pairwise different}}} H_{m_1}\cdots H_{m_s}. \tag{4.29}$$

We use the following strategy to bound the norms of these terms: (i) bound the norm of a sum of terms by summing the norms of each term; (ii) bound the norm of a product of terms by multiplying the norms of each term; (iii) bound the norm of each summand by Λ; and (iv) replace λ by $|\lambda|$. Applying this strategy, we find that the norm of the sth-order term is at most $(\Gamma\Lambda|\lambda|)^s/s!$, where the nondegenerate term contributes precisely $\Gamma(\Gamma-1)\cdots(\Gamma-s+1)(\Lambda|\lambda|)^s/s!$. Taking the difference gives the desired bound (4.26).

According to Lemma 23, the sth-order nondegenerate term of the average

evolution is

$$\frac{[(q_1 + \cdots + q_\kappa)\lambda]^s}{s!} \sum_{\substack{m_1,\ldots,m_s \\ \text{pairwise different}}} H_{m_1} \cdots H_{m_s}. \tag{4.30}$$

Following the same strategy as for $\mathscr{V}(\lambda)$ and also upper bounding the norm of each q_k by 1 as part of step (iv), we find that the norm of this term is at most

$$\frac{(\kappa\Lambda|\lambda|)^s}{s!}\Gamma(\Gamma-1)\cdots(\Gamma-s+1). \tag{4.31}$$

It remains to find an upper bound for the entire sth-order term of the average evolution. To this end, we start with the average evolution (4.14) and apply the following strategy: (i′) replace each summand of the Hamiltonian by Λ; (ii′) replace each q_k by 1 and each λ by $|\lambda|$; and (iii′) expand all exponentials into their Taylor series and extract the sth-order term. In other words, we extract the sth-order term of $\sum_{\sigma\in\mathrm{Sym}(\Gamma)} \exp(\kappa\Gamma\Lambda|\lambda|)/\Gamma!$ to get

$$\frac{(\kappa\Gamma\Lambda|\lambda|)^s}{s!}. \tag{4.32}$$

The equivalence of strategies (i)–(iv) and (i′)–(iii′) can be seen from [34, Eq. (57)]. Finally, taking the difference between (4.32) and (4.31) gives the desired bound (4.27). □

Proof of Lemma 25. We first prove a stronger bound, namely that the sth-order

error is at most

$$\begin{cases} 0 & 0 \le s \le 2k, \\ 2\frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{s!}[\Gamma^s - \Gamma(\Gamma-1)\cdots(\Gamma-s+1)] & 2k < s \le \Gamma, \\ 2\frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{s!}\Gamma^s & s > \Gamma. \end{cases} \tag{4.33}$$

The first and third cases in this expression are straightforward. The formula $\mathscr{S}_{2k}^{\sigma}$ is exact for terms with order $0 \le s \le 2k$ (this is what it means for the formula to have order $2k$), so the error is zero in this case. When $s > \Gamma$, the randomization lemma is not applicable and the error can be bounded as in [34, Proof of Proposition F.3].

To handle the remaining case $2k < s \le \Gamma$, we apply Lemma 26 with $\kappa = 2 \cdot 5^{k-1}$. This choice of κ follows from the definition of the $(2k)$th-order formula (2.14). The norm of the sth-order degenerate terms can be upper bounded by

$$\frac{(\Lambda|\lambda|)^s}{s!}\big[\Gamma^s - \Gamma(\Gamma-1)\cdots(\Gamma-s+1)\big] + \frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{s!}\big[\Gamma^s - \Gamma(\Gamma-1)\cdots(\Gamma-s+1)\big]. \tag{4.34}$$

According to Corollary 24, the sth-order nondegenerate term of (4.24) cancels, which proves (4.33) for $2k < s \le \Gamma$.

To finish the proof, we need a unified error expression for order $s > 2k$.

When $2k < s \leq \Gamma$, we have

$$\begin{aligned}
&\Gamma^s - \Gamma(\Gamma-1)\cdots(\Gamma-s+1) \\
&\quad = \#\big\{(l_1,\ldots,l_s) \in [\Gamma]^s\big\} - \#\big\{(l_1,\ldots,l_s) \in [\Gamma]^s : \forall \iota,\gamma,\ l_\iota \neq l_\gamma\big\} \\
&\quad = \#\big\{(l_1,\ldots,l_s) \in [\Gamma]^s\big\} - \#\bigcap_{\iota<\gamma}\big\{(l_1,\ldots,l_s) \in [\Gamma]^s :\ l_\iota \neq l_\gamma\big\} \\
&\quad = \#\bigcup_{\iota<\gamma}\big\{(l_1,\ldots,l_s) \in [\Gamma]^s :\ l_\iota = l_\gamma\big\} \\
&\quad \leq \binom{s}{2}\Gamma^{s-1},
\end{aligned} \tag{4.35}$$

with $\#\{\cdot\}$ denoting the size of a set and $[\Gamma] := \{1,\ldots,\Gamma\}$, where the inequality follows from the union bound. Therefore, we have

$$\begin{aligned}
2\frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{s!}[\Gamma^s - \Gamma(\Gamma-1)\cdots(\Gamma-s+1)] &\leq \frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{s!}s(s-1)\Gamma^{s-1} \\
&= \frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{(s-2)!}\Gamma^{s-1}.
\end{aligned} \tag{4.36}$$

If $s > \Gamma \in \mathbb{N}$, we have $s(s-1) \geq (\Gamma+1)\Gamma \geq 2\Gamma$ and

$$2\frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{s!}\Gamma^s \leq \frac{(2\cdot 5^{k-1}\Lambda|\lambda|)^s}{(s-2)!}\Gamma^{s-1}. \tag{4.37}$$

This completes the proof. □

We also use the following standard tail bound on the exponential function [34, Lemma F.2].

Lemma 27. *For any $x \in \mathbb{C}$ and $\kappa \in \mathbb{N}$, we have*

$$\left| \sum_{s=\kappa}^{\infty} \frac{x^s}{s!} \right| \leq \frac{|x|^\kappa}{\kappa!} \exp(|x|). \tag{4.38}$$

We now establish the main theorem, which upper bounds the error of a higher-order randomized product formula.

Theorem 28 (Randomized higher-order error bound)**.** *Let $\{H_\gamma\}_{\gamma=1}^{\Gamma}$ be Hermitian matrices. Let*

$$\mathscr{V}(-it) := \exp\left(-it \sum_{\gamma=1}^{\Gamma} H_\gamma\right) \tag{4.39}$$

be the evolution induced by the Hamiltonian $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ for time t. For any permutation $\sigma \in \mathrm{Sym}(\Gamma)$, define the permuted $(2k)$th-order formula recursively by

$$\begin{aligned} \mathscr{S}_2^\sigma(\lambda) &:= \prod_{\gamma=1}^{\Gamma} \exp\left(\frac{\lambda}{2} H_{\sigma(\gamma)}\right) \prod_{\gamma=\Gamma}^{1} \exp\left(\frac{\lambda}{2} H_{\sigma(\gamma)}\right) \\ \mathscr{S}_{2k}^\sigma(\lambda) &:= [\mathscr{S}_{2k-2}^\sigma(p_k\lambda)]^2 \mathscr{S}_{2k-2}^\sigma((1-4p_k)\lambda)[\mathscr{S}_{2k-2}^\sigma(p_k\lambda)]^2, \end{aligned} \tag{4.40}$$

with $p_k := 1/(4 - 4^{1/(2k-1)})$ for $k > 1$. Let $r \in \mathbb{N}$ and $\Lambda := \max_\gamma \|H_\gamma\|$. Then

$$\begin{aligned} &\left\| \mathcal{V}(-it) - \left(\frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(()\Gamma)} \mathcal{S}_{2k}^\sigma(-it/r) \right)^r \right\|_\diamond \\ &\quad \leq 4 \frac{(2 \cdot 5^{k-1} \Lambda |t| \Gamma)^{4k+2}}{\big((2k+1)!\big)^2 r^{4k+1}} \exp\left(4 \cdot 5^{k-1} \frac{\Lambda |t| \Gamma}{r}\right) \\ &\qquad + 2 \frac{(2 \cdot 5^{k-1} \Lambda |t|)^{2k+1} \Gamma^{2k}}{(2k-1)! r^{2k}} \exp\left(2 \cdot 5^{k-1} \frac{\Lambda |t| \Gamma}{r}\right) \end{aligned} \tag{4.41}$$

where, for $\lambda = -it$, we associate quantum channels $\mathcal{V}(\lambda)$ and $\mathcal{S}_{2k}^{\sigma}(\lambda)$ with the unitaries $\mathscr{V}(\lambda)$ and $\mathscr{S}_{2k}^{\sigma}(\lambda)$, respectively.

Proof. We first prove that

$$\begin{aligned}\left\| \mathcal{V}(\lambda) - \frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathcal{S}_{2k}^{\sigma}(\lambda) \right\|_{\diamond} \\ \leq 4 \frac{(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma)^{4k+2}}{\big((2k+1)!\big)^2} \exp\big(4 \cdot 5^{k-1} \Lambda |\lambda| \Gamma\big) \\ + 2 \frac{(2 \cdot 5^{k-1} \Lambda |\lambda|)^{2k+1} \Gamma^{2k}}{(2k-1)!} \exp\big(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma\big).\end{aligned} \tag{4.42}$$

To this end, note that the sth-order error of $\mathscr{V}(\lambda) - \mathscr{S}_{2k}^{\sigma}(\lambda)$ is at most

$$\begin{cases} 0 & 0 \leq s \leq 2k, \\ \frac{2(2 \cdot 5^{k-1} \Lambda |\lambda|)^s}{s!} \Gamma^s & s > 2k \end{cases} \tag{4.43}$$

(as before, this follows as in [34, Proof of Proposition F.3]). Thus Lemma 27 gives

$$\|\mathscr{V}(\lambda) - \mathscr{S}_{2k}^{\sigma}(\lambda)\| \leq 2 \frac{(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma)^{2k+1}}{(2k+1)!} \exp\big(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma\big). \tag{4.44}$$

On the other hand, Lemma 25 implies that the sth-order error of $\mathscr{V}(\lambda)$ $- \frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathscr{S}_{2k}^{\sigma}(\lambda)$ is at most

$$\begin{cases} 0 & 0 \leq s \leq 2k, \\ \frac{(2 \cdot 5^{k-1} \Lambda |\lambda|)^s}{(s-2)!} \Gamma^{s-1} & s > 2k, \end{cases} \tag{4.45}$$

so again Lemma 27 gives

$$\left\| \mathscr{V}(\lambda) - \frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathscr{S}_{2k}^{\sigma}(\lambda) \right\| \leq \frac{(2 \cdot 5^{k-1} \Lambda |\lambda|)^{2k+1} \Gamma^{2k}}{(2k-1)!} \exp\bigl(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma\bigr). \quad (4.46)$$

Equation (4.42) now follows from Lemma 21 by setting

$$\begin{aligned} a &= 2 \frac{(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma)^{2k+1}}{(2k+1)!} \exp\bigl(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma\bigr), \\ b &= \frac{(2 \cdot 5^{k-1} \Lambda |\lambda|)^{2k+1} \Gamma^{2k}}{(2k-1)!} \exp\bigl(2 \cdot 5^{k-1} \Lambda |\lambda| \Gamma\bigr). \end{aligned} \quad (4.47)$$

To simulate the evolution for time t, we divide it into r segments. The error within each segment is obtained from (4.42) by setting $\lambda = -it/r$. Then subadditivity of the diamond norm distance gives

$$\left\| \mathcal{V}(-it) - \left(\frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathcal{S}_{2k}^{\sigma}\bigl(-it/r\bigr) \right)^{r} \right\|_{\diamond} \leq r \left\| \mathcal{V}(-it/r) - \frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathcal{S}_{2k}^{\sigma}\bigl(-it/r\bigr) \right\|_{\diamond}, \quad (4.48)$$

which completes the proof. □

4.4 Algorithm performance and comparisons

We now analyze the complexity of our randomized product formula algorithm. Assume that $k \in \mathbb{N}$ is fixed, $\Lambda = \mathcal{O}(1)$ is constant, and $r > t\Gamma$. By Theorem 28, the asymptotic error of the $(2k)$th-order randomized product formula

is

$$\left\| \mathcal{V}(-it) - \left(\frac{1}{\Gamma!} \sum_{\sigma \in \mathrm{Sym}(\Gamma)} \mathcal{S}_{2k}^{\sigma}\big(- it/r \big) \right)^{r} \right\|_{\diamond} \leq \mathcal{O}\left(\frac{(t\Gamma)^{4k+2}}{r^{4k+1}} + \frac{t^{2k+1}\Gamma^{2k}}{r^{2k}} \right). \tag{4.49}$$

To guarantee that the simulation error is at most ϵ, we upper bound the right-hand side of (4.49) by ϵ and solve for r. We find that it suffices to use

$$\begin{aligned} r_{2k}^{\mathrm{rand}} &= \max\left\{ \mathcal{O}\left(\frac{(t\Gamma)^{\frac{4k+2}{4k+1}}}{\epsilon^{\frac{1}{4k+1}}} \right), \mathcal{O}\left(\frac{t^{\frac{2k+1}{2k}}\Gamma}{\epsilon^{\frac{1}{2k}}} \right) \right\} \\ &= \max\left\{ \mathcal{O}\left(t\Gamma \left(\frac{t\Gamma}{\epsilon} \right)^{\frac{1}{4k+1}} \right), \mathcal{O}\left(t\Gamma \left(\frac{t}{\epsilon} \right)^{\frac{1}{2k}} \right) \right\} \end{aligned} \tag{4.50}$$

steps, giving a simulation algorithm with

$$g_{2k}^{\mathrm{rand}} = \mathcal{O}(\Gamma r_{2k}^{\mathrm{rand}}) = \max\left\{ \mathcal{O}\left(t\Gamma^2 \left(\frac{t\Gamma}{\epsilon} \right)^{\frac{1}{4k+1}} \right), \mathcal{O}\left(t\Gamma^2 \left(\frac{t}{\epsilon} \right)^{\frac{1}{2k}} \right) \right\} \tag{4.51}$$

elementary gates.

For comparison, the error in the $(2k)$th-order deterministic formula algorithm is at most [34, Proposition F.4]

$$\left\| \mathscr{V}(-it) - \left[\mathscr{S}_{2k}(-it/r) \right]^{r} \right\| \leq \mathcal{O}\left(\frac{(t\Gamma)^{2k+1}}{r^{2k}} \right). \tag{4.52}$$

While this bound quantifies the simulation error in terms of the spectral-norm distance, it can easily be adapted to the diamond-norm distance using either Lemma 21 or [15, Lemma 7]. This translation introduces only constant-factor

overhead, so we have

$$\left\|\mathcal{V}(-it) - \left[\mathcal{S}_{2k}(-it/r)\right]^r\right\|_\diamond \leq \mathcal{O}\left(\frac{(t\Gamma)^{2k+1}}{r^{2k}}\right). \tag{4.53}$$

Therefore, the number of segments that suffice to ensure error at most ϵ satisfies

$$r_{2k}^{\mathrm{det}} = \mathcal{O}\left(t\Gamma\left(\frac{t\Gamma}{\epsilon}\right)^{\frac{1}{2k}}\right), \tag{4.54}$$

giving an algorithm with

$$g_{2k}^{\mathrm{det}} = \mathcal{O}(\Gamma r_{2k}^{\mathrm{det}}) = \mathcal{O}\left(t\Gamma^2\left(\frac{t\Gamma}{\epsilon}\right)^{\frac{1}{2k}}\right) \tag{4.55}$$

elementary gates. Comparing to (4.51), we see that the randomized product formula strictly improves the complexity as a function of Γ. Indeed, the $(2k)$th-order randomized approach either provides an improvement with respect to all parameters of interest over the $(2k)$th order deterministic approach (if the first term of (4.51) obtains the maximum), or has better dependence on the number of terms in the Hamiltonian than any deterministic formula (if the second term dominates).

We can also compare our result to the commutator bound of [34], which depends on the specific structure of the Hamiltonian. For concreteness, we consider a one-dimensional nearest-neighbor Heisenberg model with a random magnetic

field, as studied in [34]. Specifically, let

$$H = \sum_{j=1}^{n} (\vec{\Sigma}_j \cdot \vec{\Sigma}_{j+1} + h_j Z_j) \tag{4.56}$$

with periodic boundary conditions (i.e., $\vec{\Sigma}_{n+1} = \vec{\Sigma}_1$), and $h_j \in [-h, h]$ chosen uniformly at random, where $\vec{\Sigma}_j = (X_j, Y_j, Z_j)$ denotes a vector of Pauli x, y, and z matrices on qubit j. The $(2k)$th-order deterministic formula with the commutator bound has error at most [34, Eq. (146)]

$$\left\| \mathcal{V}(-it) - \left[\mathcal{S}_{2k}\left(-it/r\right) \right]^r \right\|_\diamond \leq \mathcal{O}\left(\frac{(t\Gamma)^{2k+2}}{r^{2k+1}} + \frac{t^{2k+1}\Gamma^{2k}}{r^{2k}} \right), \tag{4.57}$$

where we have again used Lemma 21 (or [15, Lemma 7]) to relate the spectral-norm distance to the diamond-norm distance. To guarantee that the simulation error is at most ϵ, it suffices to choose

$$\begin{aligned} r_{2k}^{\text{comm}} &= \max\left\{ \mathcal{O}\left(\frac{(t\Gamma)^{\frac{2k+2}{2k+1}}}{\epsilon^{\frac{1}{2k+1}}} \right), \mathcal{O}\left(\frac{t^{\frac{2k+1}{2k}}\Gamma}{\epsilon^{\frac{1}{2k}}} \right) \right\} \\ &= \max\left\{ \mathcal{O}\left(t\Gamma \left(\frac{t\Gamma}{\epsilon} \right)^{\frac{1}{2k+1}} \right), \mathcal{O}\left(t\Gamma \left(\frac{t}{\epsilon} \right)^{\frac{1}{2k}} \right) \right\} \end{aligned} \tag{4.58}$$

segments, giving an algorithm with

$$g_{2k}^{\text{comm}} = \mathcal{O}(\Gamma r_{2k}^{\text{comm}}) = \max\left\{ \mathcal{O}\left(t\Gamma^2 \left(\frac{t\Gamma}{\epsilon} \right)^{\frac{1}{2k+1}} \right), \mathcal{O}\left(t\Gamma^2 \left(\frac{t}{\epsilon} \right)^{\frac{1}{2k}} \right) \right\} \tag{4.59}$$

elementary gates. Comparing to the corresponding bound (4.51) for randomized

product formulas, we see that the only difference is that the exponent $1/(2k+1)$ for the commutator bound becomes $1/(4k+1)$ in the randomized case. Thus the randomized approach can provide a slightly faster algorithm despite using less information about the structure of the Hamiltonian. More specifically, the relationship between t and Γ determines whether the randomized approach offers an improvement. If $t = \Omega(\Gamma^{2k})$, then the second term of (4.59) achieves the maximum, and both approaches have asymptotic complexity $\mathcal{O}\big(t\Gamma^2\big(\frac{t}{\epsilon}\big)^{\frac{1}{2k}}\big)$. However, if $t = o(\Gamma^{2k})$, then the randomized formula is advantageous.

4.5 Empirical performance

While randomization provides a useful theoretical handle for establishing better provable bounds, those bounds may still be far from tight. As aforementioned, our original motivation for considering randomization was the observation that product formulas appear to perform dramatically better in practice than the best available proven bounds would suggest. To investigate the empirical behavior of product formulas, we numerically evaluate their performance for simulations of the Heisenberg model (4.56) with $t = n$ and $h = 1$, targeting error $\epsilon = 10^{-3}$, as previously considered in [34]. We collect data for the first-, fourth-, and sixth-order formulas as the latter two orders have the best performance in practice for small n and the first-order formula offers a qualitatively better theoretical improvement.

For the deterministic formula, we order the operators of the Hamiltonian in

the same way as [34], namely

$$X_1X_2,\ldots,X_{n-1}X_n,X_nX_1,\ Y_1Y_2,\ldots,Y_{n-1}Y_n,Y_nY_1,\ Z_1Z_2,\ldots,Z_{n-1}Z_n,Z_nZ_1,$$
$$Z_1,\ldots,Z_n. \tag{4.60}$$

We compute the error in terms of the spectral-norm distance and convert it to the diamond-norm distance using Lemma 7 of [15] (i.e., we multiply by 2). To analyze the randomized formula, we would like to numerically evaluate the diamond-norm distances

$$\left\|\mathcal{V}(-it)-\frac{1}{2^r}\left(\mathcal{S}_1(-it/r)+\mathcal{S}_1^{\mathrm{rev}}(-it/r)\right)^r\right\|_\diamond \tag{4.61}$$

and

$$\left\|\mathcal{V}(-it)-\left(\frac{1}{\Gamma!}\sum_{\sigma\in\mathrm{Sym}(\Gamma)}\mathcal{S}_{2k}^{\sigma}(-it/r)\right)^r\right\|_\diamond. \tag{4.62}$$

While the diamond norm can be computed using a semidefinite program [105], direct computation is prohibitive as the channel contains $(\Gamma!)^r$ Kraus operators. Instead, we use Lemma 21 to estimate the error. We randomly choose the ordering of the summands in each of the r segments, exponentiate each individual operator, and construct a unitary operator by concatenating the exponentials according to the given product formula. We follow this procedure to obtain a Monte Carlo

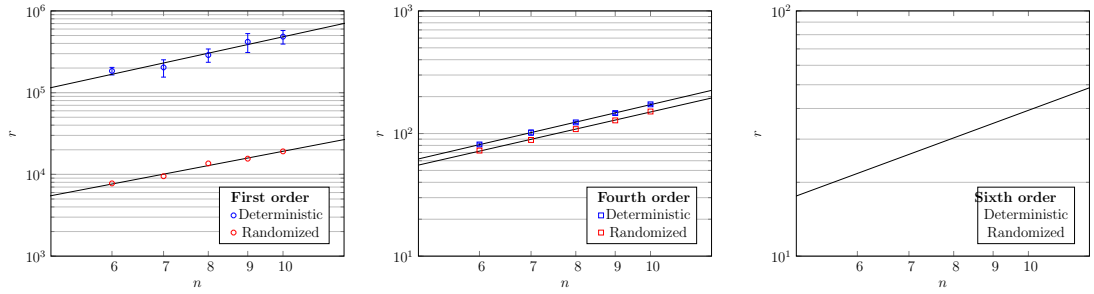

Figure 4.1: Comparison of the values of r between deterministic and randomized product formulas. Error bars are omitted when they are negligibly small on the plot. Straight lines show power-law fits to the data.

estimate of the average error

$$\left\| \mathscr{V}(-it) - \frac{1}{\mu} \sum_{m=1}^{\mu} \mathscr{S}_{2k}^{\sigma_{m,r}}\left(-it/r\right) \cdots \mathscr{S}_{2k}^{\sigma_{m,1}}\left(-it/r\right) \right\| \tag{4.63}$$

for the $(2k)$th-order formula and similarly for the first-order case. Here, μ is the number of samples in the Monte Carlo estimation, which can be increased to get more accurate estimate. In practice, we find that it suffices to take only three samples, as the standard deviations are already negligibly small (about 10^{-5}). We then invoke Lemma 21 to bound the diamond-norm error in (4.62). To the extent that the bound of Lemma 21 is loose, we expect the empirical performance to be better in practice.

Using five randomly generated instances for each value of n, we apply binary search to determine the smallest number of segments r that suffices to give error at most 10^{-3}. Figure 4.1 shows the resulting data for the first-, fourth-, and sixth-order formulas, which are well-approximated by power laws. Fitting the data, we

estimate that

$$r_1^{\text{remp}} = 300.0n^{1.806} \qquad r_4^{\text{remp}} = 5.458n^{1.439} \qquad r_6^{\text{remp}} = 2.804n^{1.152} \tag{4.64}$$

segments should suffice to give error at most 10^{-3}. We thus observe that the empirical complexity of the randomized algorithm is still significantly better than the provable performance

$$r_1^{\text{rand}} = \mathcal{O}(n^3) \qquad r_4^{\text{rand}} = \mathcal{O}(n^{2.25}) \qquad r_6^{\text{rand}} = \mathcal{O}(n^{2.17}). \tag{4.65}$$

For comparison, analogous empirical fits for deterministic formulas give the comparable values

$$r_1^{\text{demp}} = 4143n^{2.066} \qquad r_4^{\text{demp}} = 5.821n^{1.471} \qquad r_6^{\text{demp}} = 2.719n^{1.160}, \tag{4.66}$$

(cf. [34, Eq. (147)], but note that we have generated new data using [15, Lemma 7] to bound the diamond-norm distance in terms of the spectral-norm distance), whereas the rigorous commutator bound gives the larger exponents [34]

$$r_1^{\text{comm}} = \mathcal{O}(n^3) \qquad r_4^{\text{comm}} = \mathcal{O}(n^{2.4}) \qquad r_6^{\text{comm}} = \mathcal{O}(n^{2.28}). \tag{4.67}$$

We see that the randomized bound offers significantly better empirical performance at first order, consistent with the observation that randomization improves

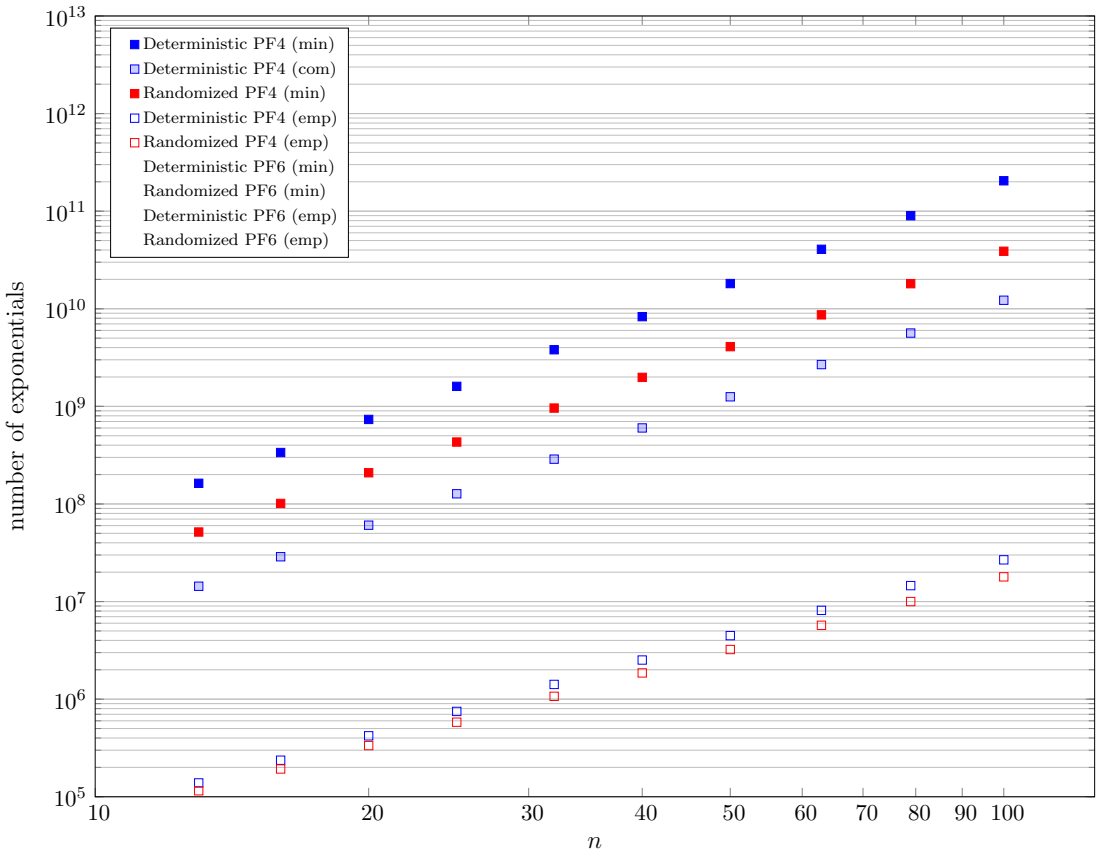

Figure 4.2: Comparison of the total number of elementary exponentials for product formula simulations of the Heisenberg model using deterministic and randomized product formulas of fourth and sixth order with both rigorous and empirical error bounds. Note that since the empirical performance of deterministic and randomized sixth-order product formulas is almost the same, the latter data points are obscured by the former.

the order of approximation in this case. The fourth-order formula slightly improves both the exponent and the constant factor. While this improvement is small, it is nevertheless notable since it involves only a minor change to the algorithm. At sixth order we see negligible improvement. Since the proven bounds give less improvement with each successive order, it is perhaps not surprising to see that the empirical performance shows similar behavior.

To illustrate the effect of using different formulas and different error bounds

to simulate larger systems, Figure 4.2 compares the cost of simulating our model system for sizes up to $n = 100$ with deterministic and randomized formulas of orders 4 and 6, using both proven error bounds and the above empirical estimates. (We omit the first-order formula since it is not competitive even at such small sizes.) We give rigorous bounds for deterministic formulas using the minimized bound of [34], and for fourth order we also show the result of using the commutator bound. We see that randomization gives a significant improvement over the deterministic formula using the minimized bound, although the commutator bound outperforms the randomized bound at the system sizes shown here. For sufficiently large n, the randomized bound gives lower complexity, but this requires a fairly large n since the difference in exponents is small and the commutator bound achieves a favorable constant prefactor. Empirical estimates of the error improve the performance by several orders of magnitude, with randomization giving a small advantage for the fourth-order formula as indicated above. However, for systems of size larger than about $n = 25$, the sixth-order bound prevails, and in this case randomization no longer offers a significant advantage.

4.6 Discussion

We have shown that randomization can be used to establish better performance for quantum simulation algorithms based on product formulas. By simply randomizing how the summands in the Hamiltonian are ordered, we introduce

terms in the average evolution that could not appear in any deterministic product formula approximation of the same order, and thereby give a more efficient algorithm. Indeed, this approach can outperform the commutator bound even though that method uses more information about the structure of the Hamiltonian. A randomized product formula simulation algorithm is not much more complicated than the corresponding deterministic formula, using only $\mathcal{O}(\Gamma \log \Gamma)$ bits of randomness per segment and no ancilla qubits. Furthermore, we showed that randomization can even offer improved empirical performance in some cases.

While randomization has allowed us to make some progress on the challenge of proving better bounds on the performance of product formulas, our strengthened bounds remain far from the apparent empirical performance. We will address this in Chapter 6 and Chapter 7 by developing a general theory for analyzing the error of product formulas. More generally, it may be of interest to investigate other scenarios in which random choices can be used to improve the analysis of quantum simulation [16, 25] and other quantum algorithms.

Chapter 5: Randomized time-dependent Hamiltonian simulation

In this chapter, we consider time-dependent Hamiltonian simulation and develop a randomized approach with L^1-norm scaling, strictly faster than existing quantum simulation algorithms. We give motivations to studying L^1-norm scaled algorithms and discuss the limitations of existing approaches in Section 5.1. In Section 5.2, we introduce a classical sampling protocol, which we call "continuous qDRIFT", to achieve the L^1-norm scaling for time-dependent Hamiltonian simulation. For the purpose of presentation, we first assume that the Hamiltonian at each time can be efficiently exponentiated and later relax this assumption in Section 5.3 by proving a universal property of our protocol. We conclude in Section 5.4 with a brief discussion of the results and some open questions.

This chapter is partly based on the following paper:

[16] Dominic W. Berry, Andrew M. Childs, Yuan Su, Xin Wang, and Nathan Wiebe, *Time-dependent Hamiltonian simulation with L^1-norm scaling*, Quantum **4** (2020), 254, arXiv:1906.07115.

5.1 L^1-norm scaling

We develop algorithms for time-dependent Hamiltonian simulation based on a simple intuition: the difficulty of simulating a quantum system should depend on the integrated norm of the Hamiltonian. To elaborate, first consider the special

case of simulating a time-independent Hamiltonian. The complexity of such a simulation depends on $t\,\|H\|$ [31], where $\|\cdot\|$ is a matrix norm that quantifies the size of the Hamiltonian. It is common to express the complexity in terms of the spectral norm, which quantifies the maximum energy of H.

In the general case where the Hamiltonian $\mathscr{H}(\tau)$ is time dependent, we expect a quantum simulation algorithm to depend on the Hamiltonian locally in time, and therefore to have complexity that scales with the integrated spectral norm $\int_0^t \mathrm{d}\tau\,\|\mathscr{H}(\tau)\|$. This is the L^1 norm of $\|\mathscr{H}(\tau)\|$ when viewed as a function of τ, so we say such an algorithm has L^1-norm scaling. Surprisingly, existing simulation algorithms fail to achieve this complexity; rather, their gate complexity scales with the worst-case cost $t\max_{\tau\in[0,t]}\|\mathscr{H}(\tau)\|$. It is therefore reasonable to question whether our intuition is correct, or if there exist faster time-dependent Hamiltonian simulation algorithms that can exploit this intuition.[1]

We answer this question by providing a faster quantum algorithm for time-dependent Hamiltonian simulation based on randomization. This algorithm has gate complexity that scales with the L^1 norm $\int_0^t \mathrm{d}\tau\,\|\mathscr{H}(\tau)\|$, in contrast to the best previous scaling of $t\max_{\tau\in[0,t]}\|\mathscr{H}(\tau)\|$. As the norm inequality $\int_0^t \mathrm{d}\tau\,\|\mathscr{H}(\tau)\| \leq t\max_{\tau\in[0,t]}\|\mathscr{H}(\tau)\|$ always holds but is not saturated in general, this algorithm provides strict speedups over existing algorithms.

[1] For the Dyson-series approach, Low and Wiebe claimed that the worst-case scaling may be avoided by a proper segmentation of the time interval [76, Section VI. A]. However, it is unclear how their analysis can be formalized to give an algorithm with complexity that scales with the L^1 norm. Instead, we propose a rescaling principle for the Schrödinger equation in [16, Section 4] and develop a rescaled Dyson-series algorithm with L^1-norm scaling.

5.2 A classical sampler of time-dependent Hamiltonians

Let $\mathscr{H}(\tau)$ be a time-dependent Hamiltonian defined for $0 \leq \tau \leq t$. We assume that $\mathscr{H}(\tau)$ is nonzero everywhere and is continuous except on a finite number of points. We further suppose that each $\mathscr{H}(\tau)$ can be directly exponentiated on a quantum computer. We denote the ideal evolution under $\mathscr{H}(\tau)$ for time $\tau \in [0, t]$ by $\mathrm{E}(t, 0) := \exp_{\mathcal{T}}\big(-i\int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau)\big)$ and represent the corresponding quantum channel as

$$\mathcal{E}(t,0)(\rho) = \mathrm{E}(t,0)\rho\mathrm{E}^{\dagger}(t,0) = \exp_{\mathcal{T}}\left(-i\int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau)\right)\rho\exp_{\mathcal{T}}^{\dagger}\left(-i\int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau)\right). \tag{5.1}$$

The high-level idea of the sampling algorithm is to approximate the ideal channel by a mixed unitary channel

$$\mathcal{U}(t,0)(\rho) := \int_0^t \mathrm{d}\tau\, p(\tau) e^{-i\frac{\mathscr{H}(\tau)}{p(\tau)}}\rho e^{i\frac{\mathscr{H}(\tau)}{p(\tau)}}, \tag{5.2}$$

where $p(\tau)$ is a probability density function defined for $0 \leq \tau \leq t$. This channel can be realized by a classical sampling protocol. With a proper choice of $p(\tau)$, this channel approximates the ideal channel and can thus be used for quantum simulation.

We begin with a full definition of $\mathcal{U}(t, 0)$. Inspired by [25], we choose $p(\tau)$ to

be biased toward those τ with large $\|\mathscr{H}(\tau)\|$. A natural choice is

$$p(\tau) := \frac{\|\mathscr{H}(\tau)\|}{\|\mathscr{H}\|_1}, \tag{5.3}$$

where

$$\|\mathscr{H}\|_1 := \int_0^t \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| \tag{5.4}$$

is the L^1 norm of $\mathscr{H}(\tau)$. Note that $\mathcal{U}(t,0)$ is a valid quantum channel (in particular, $p(\tau)$ can never be zero). Furthermore, it can be implemented with unit cost: for any input state ρ, we randomly sample a value τ according to $p(\tau)$ and perform $e^{-i\mathscr{H}(\tau)/p(\tau)}$. Note also that $\mathscr{H}(\tau)/p(\tau)$ in the exponential implicitly depends on t. Indeed, $\|\mathscr{H}\|_1$ includes an integral over time, so $p(\tau)$ decreases with the total evolution time t. We call this classical sampling protocol and the channel it implements "continuous qDRIFT".

This protocol assumes that the spectral norm $\|\mathscr{H}(\tau)\|$ is known a priori and that we can efficiently sample from the distribution $p(\tau)$. In practice, it is often easier to obtain a spectral-norm upper bound $\Lambda(\tau) \geq \|\mathscr{H}(\tau)\|$. Such an upper bound can also be used to implement continuous qDRIFT, provided that it has only finitely many discontinuities. Specifically, we define

$$p_\Lambda(\tau) := \frac{\Lambda(\tau)}{\|\Lambda\|_1} \tag{5.5}$$

with $\|\Lambda\|_1 := \int_0^t \mathrm{d}\tau \Lambda(\tau)$, so $p_\Lambda(\tau)$ is a probability density function. Using p_Λ to

implement continuous qDRIFT, we obtain the channel

$$\mathcal{U}_\Lambda(t,0)(\rho) := \int_0^t \mathrm{d}\tau\, p_\Lambda(\tau) e^{-i\frac{\mathscr{H}(\tau)}{p_\Lambda(\tau)}} \rho e^{i\frac{\mathscr{H}(\tau)}{p_\Lambda(\tau)}}, \tag{5.6}$$

whose analysis is similar to that presented below. For readability, we assume that we can efficiently sample from $p(\tau) = \|\mathscr{H}(\tau)\| / \|\mathscr{H}\|_1$ and we analyze $\mathcal{U}(t,0)$.

We show that continuous qDRIFT approximates the ideal channel with error that depends on the L^1-norm.

Theorem 29 (L^1-norm error bound for continuous qDRIFT, short-time version)**.** *Let $\mathscr{H}(\tau)$ be a time-dependent Hamiltonian defined for $0 \leq \tau \leq t$; assume it is continuous except on a finite number of points and nonzero everywhere. Define $\mathrm{E}(t,0) = \exp_{\mathcal{T}}\big(-i\int_0^t \mathrm{d}\tau \mathscr{H}(\tau)\big)$ and let $\mathcal{E}(t,0)(\cdot) = \mathrm{E}(t,0)(\cdot)\mathrm{E}^\dagger(t,0)$ be the corresponding quantum channel. Let $\mathcal{U}(t,0)$ be the continuous qDRIFT channel*

$$\mathcal{U}(t,0)(\rho) = \int_0^t \mathrm{d}\tau\, p(\tau) e^{-i\frac{\mathscr{H}(\tau)}{p(\tau)}} \rho e^{i\frac{\mathscr{H}(\tau)}{p(\tau)}}, \tag{5.7}$$

where $p(\tau) = \|\mathscr{H}(\tau)\| / \|\mathscr{H}\|_1$. Then

$$\|\mathcal{E}(t,0) - \mathcal{U}(t,0)\|_\diamond \leq 4\,\|\mathscr{H}\|_1^2\,. \tag{5.8}$$

To prove this theorem, we need a formula that computes the rate at which the evolution operator changes when the Hamiltonian is scaled. To illustrate the

idea, consider the degenerate case where the Hamiltonian H is time independent. Then the evolution under H for time t is given by e^{-itH}. A direct calculation shows that

$$\frac{\mathrm{d}}{\mathrm{d}s} e^{-itsH} = -itHe^{-itsH}, \tag{5.9}$$

so the rate is $-itHe^{-itsH}$ in the time-independent case. This calculation becomes significantly more complicated for a time-dependent Hamiltonian. The following lemma gives an explicit formula for

$$\frac{\mathrm{d}}{\mathrm{d}s} \exp_{\mathcal{T}}\left(-i \int_0^t \mathrm{d}\tau \, sH(\tau)\right). \tag{5.10}$$

We sketch the proof of this formula for completeness, but refer the reader to [42, p. 35] for mathematical justifications that are beyond the scope of this paper.

Lemma 30 (Hamiltonian scaling)**.** *Let $\mathscr{H}(\tau)$ be a time-dependent Hamiltonian defined for $0 \leq \tau \leq t$ and assume it has finitely many discontinuities. Denote $\mathrm{E}_s(t, v) = \exp_{\mathcal{T}}\big(-i \int_v^t \mathrm{d}\tau \, s\mathscr{H}(\tau)\big)$. Then,*

$$\frac{\mathrm{d}}{\mathrm{d}s} \mathrm{E}_s(t, v) = \int_v^t \mathrm{d}\tau \, \mathrm{E}_s(t, \tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau, v). \tag{5.11}$$

Proof sketch. We first consider the special case where $\mathscr{H}(\tau)$ is continuous in τ. We invoke the variation-of-parameters formula [68, Theorem 4.9] to construct the claimed integral representation for $\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t, v)$. To this end, we need to find a

differential equation satisfied by $\frac{\mathrm{d}}{\mathrm{d}t}\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v)$ and the corresponding initial condition $\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v)\big|_{t=v}$. We differentiate Schrödinger equation $\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{E}_s(t,v) = -is\mathscr{H}(t)\mathrm{E}_s(t,v)$ with respect to s to get

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v) = -is\mathscr{H}(t)\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v) - i\mathscr{H}(t)\mathrm{E}_s(t,v). \tag{5.12}$$

Invoking the variation-of-parameters formula, we find an integral representation

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v) &= \mathrm{E}_s(t,v)\cdot\left[\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v)\Big|_{t=v}\right] + \mathrm{E}_s(t,v)\int_v^t \mathrm{d}\tau\,\mathrm{E}_s^\dagger(\tau,v)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,v)\\ &= \mathrm{E}_s(t,v)\cdot\left[\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v)\Big|_{t=v}\right] + \int_v^t \mathrm{d}\tau\,\mathrm{E}_s(t,\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,v).\end{aligned} \tag{5.13}$$

It thus remains to find the initial condition $\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v)\big|_{t=v}$.

We start from the Schrödinger equation $\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{E}_s(t,v) = -is\mathscr{H}(t)\mathrm{E}_s(t,v)$ and apply the fundamental theorem of calculus with initial condition $\mathrm{E}_s(v,v) = I$, obtaining the integral representation

$$\mathrm{E}_s(t,v) = I - is\int_v^t \mathrm{d}\tau\,\mathscr{H}(\tau)\mathrm{E}_s(\tau,v). \tag{5.14}$$

Differentiating this equation with respect to s gives

$$\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v) = -i\int_v^t \mathrm{d}\tau\,\mathscr{H}(\tau)\mathrm{E}_s(t,v) - is\int_v^t \mathrm{d}\tau\,\mathscr{H}(\tau)\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(\tau,v), \tag{5.15}$$

which implies

$$\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v)\Big|_{t=v} = 0. \tag{5.16}$$

Combining (5.13) and (5.16) establishes the claimed integral representation for $\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v)$.

Now consider the case where $\mathscr{H}(\tau)$ is piecewise continuous with one discontinuity at $t_1 \in [v,t]$. We use the multiplicative property to break the evolution at t_1, so that each subevolution is generated by a continuous Hamiltonian. We have

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,v) &= \frac{\mathrm{d}}{\mathrm{d}s}\big[\mathrm{E}_s(t,t_1)\mathrm{E}_s(t_1,v)\big] \\
&= \frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,t_1)\cdot\mathrm{E}_s(t_1,v) + \mathrm{E}_s(t,t_1)\cdot\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t_1,v) \\
&= \int_{t_1}^{t}\mathrm{d}\tau\,\mathrm{E}_s(t,\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,t_1)\cdot\mathrm{E}_s(t_1,v) \\
&\quad + \mathrm{E}_s(t,t_1)\cdot\int_{0}^{t_1}\mathrm{d}\tau\,\mathrm{E}_s(t_1,\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,v) \\
&= \int_{t_1}^{t}\mathrm{d}\tau\,\mathrm{E}_s(t,\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,v) \\
&\quad + \int_{v}^{t_1}\mathrm{d}\tau\,\mathrm{E}_s(t,\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,v) \\
&= \int_{v}^{t}\mathrm{d}\tau\,\mathrm{E}_s(t,\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,v).
\end{aligned} \tag{5.17}$$

The general case of finitely many discontinuities follows by induction. □

Note that our argument implicitly assumes the existence of the derivatives and that we can interchange the order of $\frac{\mathrm{d}}{\mathrm{d}s}$ and $\frac{\mathrm{d}}{\mathrm{d}t}$. A rigorous justification of these assumptions is beyond the scope of the paper; we refer the reader to [42, p.

35] for details.

Proof of Theorem 29. Define two parametrized quantum channels

$$\mathcal{E}_s(t,0)(\rho) = \mathrm{E}_s(t,0)\rho\mathrm{E}_s^\dagger(t,0), \qquad \mathcal{U}_s(t,0)(\rho) = \int_0^t \mathrm{d}\tau\, p(\tau) e^{-is\frac{\mathscr{H}(\tau)}{p(\tau)}} \rho e^{is\frac{\mathscr{H}(\tau)}{p(\tau)}} \tag{5.18}$$

and observe that

$$\mathcal{E}_0(t,0)(\rho) = \rho, \quad \mathcal{E}_1(t,0)(\rho) = \mathcal{E}(t,0)(\rho), \quad \mathcal{U}_0(t,0)(\rho) = \rho, \quad \mathcal{U}_1(t,0)(\rho) = \mathcal{U}(t,0)(\rho). \tag{5.19}$$

To bound the diamond-norm error $\|\mathcal{E}_1(t,0) - \mathcal{U}_1(t,0)\|_\diamond$, we should take a state σ on the joint system of the original register and an ancilla register with the same dimension and upper bound $\|(\mathcal{E}_1(t,0)\otimes\mathbb{1})(\sigma) - (\mathcal{U}_1(t,0)\otimes\mathbb{1})(\sigma)\|_{\mathrm{tr}}$. For readability, we instead show how to bound the error $\|\mathcal{E}_1(t,0)(\rho) - \mathcal{U}_1(t,0)(\rho)\|_{\mathrm{tr}}$, but the derivation works in exactly the same way for the distance $\|(\mathcal{E}_1(t,0)\otimes\mathbb{1})(\sigma) - (\mathcal{U}_1(t,0)\otimes\mathbb{1})(\sigma)\|_{\mathrm{tr}}$ and the resulting bound is the same.

Invoking Lemma 30, we have

$$\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{E}_s(t,0)\Big|_{s=0} = \int_0^t \mathrm{d}\tau\, \mathrm{E}_s(t,\tau)\Big|_{s=0}\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_s(\tau,0)\Big|_{s=0} = -i\int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau). \tag{5.20}$$

Thus, the first derivatives of $\mathcal{E}_s(t,0)(\rho)$ and $\mathcal{U}_s(t,0)(\rho)$ at $s=0$ agree with each

other:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}s}\mathcal{E}_s(t,0)(\rho)\Big|_{s=0} &= \Big[-i\int_0^t \mathrm{d}\tau\,\mathscr{H}(\tau),\rho\Big] \\ &= \int_0^t \mathrm{d}\tau\,p(\tau)\Big[-i\frac{\mathscr{H}(\tau)}{p(\tau)},\rho\Big] = \frac{\mathrm{d}}{\mathrm{d}s}\mathcal{U}_s(t,0)(\rho)\Big|_{s=0}.\end{aligned} \tag{5.21}$$

Applying the fundamental theorem of calculus twice, we obtain

$$\begin{aligned}\mathcal{E}_1(t,0)(\rho)-\mathcal{U}_1(t,0)(\rho) &= \big(\mathcal{E}_1(t,0)(\rho)-\mathcal{E}_0(t,0)(\rho)\big)-\big(\mathcal{U}_1(t,0)(\rho)-\mathcal{U}_0(t,0)(\rho)\big) \\ &= \int_0^1 \mathrm{d}s\int_0^s \mathrm{d}v\;\frac{\mathrm{d}^2}{\mathrm{d}v^2}\big[\mathcal{E}_v(t,0)(\rho)-\mathcal{U}_v(t,0)(\rho)\big] \\ &= \int_0^1 \mathrm{d}s\int_0^s \mathrm{d}v\;\Bigg\{\frac{\mathrm{d}^2}{\mathrm{d}v^2}\mathrm{E}_v(t,0)\cdot\rho\cdot\mathrm{E}_v^\dagger(t,0) \\ &\quad + 2\frac{\mathrm{d}}{\mathrm{d}v}\mathrm{E}_v(t,0)\cdot\rho\cdot\frac{\mathrm{d}}{\mathrm{d}v}\mathrm{E}_v^\dagger(t,0)+\mathrm{E}_v(t,0)\cdot\rho\cdot\frac{\mathrm{d}^2}{\mathrm{d}v^2}\mathrm{E}_v^\dagger(t,0) \\ &\quad -\int_0^t \mathrm{d}\tau\,p(\tau)e^{-iv\frac{\mathscr{H}(\tau)}{p(\tau)}}\Big[-i\frac{\mathscr{H}(\tau)}{p(\tau)},\Big[-i\frac{\mathscr{H}(\tau)}{p(\tau)},\rho\Big]\Big]e^{iv\frac{\mathscr{H}(\tau)}{p(\tau)}}\Bigg\}.\end{aligned} \tag{5.22}$$

By properties of the Schatten norms and the definition $p(\tau) = \|\mathscr{H}(\tau)\| / \|\mathscr{H}\|_1$, we find that

$$\begin{aligned}\|\mathcal{E}_1(t,0)(\rho)-\mathcal{U}_1(t,0)(\rho)\|_{\mathrm{tr}} \le \int_0^1 \mathrm{d}s\int_0^s \mathrm{d}v\;\Bigg\{2\Big\|\frac{\mathrm{d}^2}{\mathrm{d}v^2}\mathrm{E}_v(t,0)\Big\|+2\Big\|\frac{\mathrm{d}}{\mathrm{d}v}\mathrm{E}_v(t,0)\Big\|^2 \\ +4\|\mathscr{H}\|_1^2\Bigg\}.\end{aligned} \tag{5.23}$$

Lemma 30 immediately yields an upper bound on $\big\|\frac{\mathrm{d}}{\mathrm{d}v}\mathrm{E}_v(t,0)\big\|$:

$$\Big\|\frac{\mathrm{d}}{\mathrm{d}v}\mathrm{E}_v(t,0)\Big\| \le \int_0^t \mathrm{d}\tau\,\|\mathscr{H}(\tau)\| = \|\mathscr{H}\|_1. \tag{5.24}$$

It thus remains to bound $\left\| \frac{\mathrm{d}^2}{\mathrm{d}v^2} \mathrm{E}_v(t,0) \right\|$.

Using Lemma 30 twice, we have

$$\begin{aligned}
\frac{\mathrm{d}^2}{\mathrm{d}v^2}\mathrm{E}_v(t,0) &= \frac{\mathrm{d}}{\mathrm{d}v}\int_0^t \mathrm{d}\tau\, \mathrm{E}_v(t,\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_v(\tau,0) \\
&= \int_0^t \mathrm{d}\tau \int_\tau^t \mathrm{d}\tau'\, \mathrm{E}_v(t,\tau')\big[-i\mathscr{H}(\tau')\big]\mathrm{E}_v(\tau',\tau)\big[-i\mathscr{H}(\tau)\big]\mathrm{E}_v(\tau,0) \\
&\quad + \int_0^t \mathrm{d}\tau\, \mathrm{E}_v(t,\tau)\big[-i\mathscr{H}(\tau)\big] \int_0^\tau \mathrm{d}\tau'\, \mathrm{E}_v(\tau,\tau')\big[-i\mathscr{H}(\tau')\big]\mathrm{E}_v(\tau',0),
\end{aligned} \tag{5.25}$$

which implies

$$\begin{aligned}
\left\| \frac{\mathrm{d}^2}{\mathrm{d}v^2}\mathrm{E}_v(\tau,0) \right\| &\le \int_0^t \mathrm{d}\tau \int_\tau^t \mathrm{d}\tau'\, \|\mathscr{H}(\tau')\|\, \|\mathscr{H}(\tau)\| + \int_0^t \mathrm{d}\tau \int_0^\tau \mathrm{d}\tau'\, \|\mathscr{H}(\tau)\|\, \|\mathscr{H}(\tau')\| \\
&= 2\,\|\mathscr{H}\|_1^2 .
\end{aligned} \tag{5.26}$$

We finally obtain the desired bound

$$\|\mathcal{E}_1(t,0)(\rho) - \mathcal{U}_1(t,0)(\rho)\|_{\mathrm{tr}} \le \int_0^1 \mathrm{d}s \int_0^s \mathrm{d}v \left[2\,\|\mathscr{H}\|_1^2 + 2\,\|\mathscr{H}\|_1^2 + 4\,\|\mathscr{H}\|_1^2\right] = 4\,\|\mathscr{H}\|_1^2 \tag{5.27}$$

as claimed. □

The above error bound works well for a short-time evolution. When t is large, in order to control the error of simulation, we divide the entire evolution into segments $[t_j, t_{j+1}]$ with $0 = t_0 < t_1 < \cdots < t_r = t$ and apply continuous qDRIFT within each. We employ a variable-time scheme to segment the evolution, so that our L^1-norm scaling result can be generalized to a long-time evolution. Specifically,

we have:

Theorem 31 (L^1-norm error bound for continuous qDRIFT, long-time version)**.** *Let $\mathscr{H}(\tau)$ be a time-dependent Hamiltonian defined for $0 \leq \tau \leq t$. Assume that it is continuous except at a finite number of points and nonzero everywhere. Define $\mathrm{E}(t,0) = \exp_{\mathcal{T}}\big(-i\int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau)\big)$ and let $\mathcal{E}(t,0)(\cdot) = \mathrm{E}(t,0)(\cdot)\mathrm{E}^\dagger(t,0)$ be the corresponding quantum channel. Let $\mathcal{U}(t,0)$ be the continuous qDRIFT channel*

$$\mathcal{U}(t,0)(\rho) = \int_0^t \mathrm{d}\tau\, p(\tau) e^{-i\frac{\mathscr{H}(\tau)}{p(\tau)}} \rho e^{i\frac{\mathscr{H}(\tau)}{p(\tau)}}, \tag{5.28}$$

where $p(\tau) = \|\mathscr{H}(\tau)\| / \|\mathscr{H}\|_1$. Then, for any positive integer r, there exists a division $0 = t_0 < t_1 < \cdots < t_r = t$, such that

$$\left\| \mathcal{E}(t,0) - \prod_{j=0}^{r-1} \mathcal{U}(t_{j+1}, t_j) \right\|_\diamond \leq 4\frac{\|\mathscr{H}\|_1^2}{r}. \tag{5.29}$$

To ensure that the simulation error is at most ϵ, it thus suffices to choose

$$r \geq 4\left\lceil \frac{\|\mathscr{H}\|_1^2}{\epsilon} \right\rceil. \tag{5.30}$$

Proof. The times $t_1, \cdots, t_{r-1}$ are selected as follows. We aim to simulate with accuracy

$$4\frac{\|\mathscr{H}\|_1^2}{r^2} \tag{5.31}$$

for each segment. To achieve this, we define $t_1, \cdots, t_{r-1}$ so that

$$\int_0^{t_1} \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| = \int_{t_1}^{t_2} \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| = \cdots = \int_{t_{r-1}}^{t_r} \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| = \frac{1}{r} \int_0^t \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| \,. \tag{5.32}$$

The existence of such times is guaranteed by the intermediate value theorem. By telescoping, we find from Theorem 29 that

$$\begin{aligned} \left\| \mathcal{E}(t,0) - \prod_{j=0}^{r-1} \mathcal{U}(t_{j+1}, t_j) \right\|_\diamond &\leq \sum_{j=0}^{r-1} \|\mathcal{U}(t_{j+1}, t_j) - \mathcal{E}(t_{j+1}, t_j)\|_\diamond \\ &\leq \sum_{j=0}^{r-1} 4 \left(\int_{t_j}^{t_{j+1}} \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| \right)^2 \\ &= 4r \left(\frac{1}{r} \int_0^t \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| \right)^2 = 4 \frac{\|\mathscr{H}\|_1^2}{r}, \end{aligned} \tag{5.33}$$

which establishes the claimed error bound. □

5.3 Universality

We now extend our above analysis to the general LC model. Recall from Section 2.3 that the Hamiltonian can be expressed as

$$\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \mathscr{H}_\gamma(\tau), \tag{5.34}$$

where each $\mathscr{H}_\gamma(\tau)$ is continuous, nonzero everywhere, and can be efficiently exponentiated on a quantum computer.

It is not hard to design a classical sampler for time-dependent Hamiltonians in the LC model. A natural choice is

$$\mathcal{U}(t,0)(\rho) := \sum_{l=1}^{\Gamma} \int_0^t \mathrm{d}\tau \, p_\gamma(\tau) e^{-i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}} \rho e^{i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}}, \tag{5.35}$$

where $p_\gamma(\tau)$ is the probability distribution

$$p_\gamma(\tau) := \frac{\|\mathscr{H}_\gamma(\tau)\|}{\|\mathscr{H}\|_{1,1}}, \tag{5.36}$$

where

$$\|\mathscr{H}\|_{1,1} := \int_0^t \mathrm{d}\tau \sum_{\gamma=1}^{\Gamma} \|\mathscr{H}_\gamma(\tau)\| . \tag{5.37}$$

To analyze the performance of this sampler, we adapt the analysis in Theorem 29 and Theorem 31, which becomes more complicated as we are now sampling a discrete-continuous probability distribution $p_\gamma(\tau)$. Fortunately, a significant amount of effort can be saved with the help of the following universal property.

Theorem 32 (Universality of continuous qDRIFT)**.** *Let $\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \mathscr{H}_\gamma(\tau)$ be a time-dependent Hamiltonian defined for $0 \leq \tau \leq t$ that is nonzero everywhere. Assume that each $\mathscr{H}_\gamma(\tau)$ is continuous and nonzero everywhere. Define the probability distribution*

$$p_\gamma(\tau) := \frac{\|\mathscr{H}_\gamma(\tau)\|}{\|\mathscr{H}\|_{1,1}}. \tag{5.38}$$

Then there exists a time-dependent Hamiltonian $\mathscr{G}(\tau)$ defined for $0 \leq \tau \leq t$ with

finitely many discontinuities, such that the following correspondence holds:

1. $\|\mathscr{G}\|_1 = \|\mathscr{H}\|_{1,1}$.

2. $\int_0^t \mathrm{d}\tau\, \mathscr{G}(\tau) = \sum_{\gamma=1}^{\Gamma} \int_0^t \mathrm{d}\tau\, \mathscr{H}_\gamma(\tau)$.

3. $\int_0^t \mathrm{d}\tau\, q(\tau) e^{-i\frac{\mathscr{G}(\tau)}{q(\tau)}} \rho e^{i\frac{\mathscr{G}(\tau)}{q(\tau)}} = \sum_{\gamma=1}^{\Gamma} \int_0^t \mathrm{d}\tau\, p_\gamma(\tau) e^{-i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}} \rho e^{i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}}$, *where we have the probability distribution* $q(\tau) := \|\mathscr{G}(\tau)\| \,/\, \|\mathscr{G}\|_1$.

Before presenting the proof, we explain how Theorem 32 can be applied to simulation in the LC model. We expect that the mixed-unitary channel

$$\sum_{\gamma=1}^{\Gamma} \int_0^t \mathrm{d}\tau\, p_\gamma(\tau) e^{-i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}} \rho e^{i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}} \tag{5.39}$$

approximates the ideal evolution with L^1-norm scaling as in Theorem 29 and Theorem 31, but direct analysis would be considerably more complicated. However, universality (Statement 3 of Theorem 32) shows that this channel is the same as $\int_0^t \mathrm{d}\tau\, q(\tau) e^{-i\frac{\mathscr{G}(\tau)}{q(\tau)}} \rho e^{i\frac{\mathscr{G}(\tau)}{q(\tau)}}$. Thus, the analysis of Section 5.2 can be applied with the help of Theorem 32.

Proof of Theorem 32. We define $\mathscr{G}(\tau)$ to be the piecewise Hamiltonian

$$\mathscr{G}(\tau) = \begin{cases} \frac{\mathscr{H}_1\left(\frac{\tau}{p_1}\right)}{p_1}, & 0 \leq \tau < p_1 t, \\ \frac{\mathscr{H}_2\left(\frac{\tau - p_1 t}{p_2}\right)}{p_2}, & p_1 t \leq \tau < (p_1 + p_2)t, \\ \quad \vdots & \\ \frac{\mathscr{H}_\Gamma\left(\frac{\tau - (p_1 + p_2 + \cdots + p_{\Gamma-1})t}{p_\Gamma}\right)}{p_\Gamma}, & (p_1 + p_2 + \cdots + p_{\Gamma-1})t \leq \tau \leq t, \end{cases} \tag{5.40}$$

where we use the abbreviation

$$p_\gamma := \|p_\gamma\|_1 = \int_0^t \mathrm{d}\tau \, p_\gamma(\tau) \tag{5.41}$$

for the marginal probability distribution. Statements 1 and 2 can both be proved by directly evaluating the integrals

$$\begin{aligned} \|\mathscr{G}\|_1 &= \int_0^{p_1 t} \mathrm{d}\tau \frac{\left\| \mathscr{H}_1\left(\frac{\tau}{p_1}\right) \right\|}{p_1} \\ &\quad + \int_{p_1 t}^{(p_1+p_2)t} \mathrm{d}\tau \frac{\left\| \mathscr{H}_2\left(\frac{\tau - p_1 t}{p_2}\right) \right\|}{p_2} \\ &\quad + \cdots + \int_{(p_1+p_2+\cdots+p_{\Gamma-1})t}^{t} \mathrm{d}\tau \frac{\left\| \mathscr{H}_\Gamma\left(\frac{\tau - (p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right) \right\|}{p_\Gamma} \\ &= \int_0^t \mathrm{d}\tau \, \|\mathscr{H}_1(\tau)\| + \int_0^t \mathrm{d}\tau \, \|\mathscr{H}_2(\tau)\| + \cdots + \int_0^t \mathrm{d}\tau \, \|\mathscr{H}_\Gamma(\tau)\| \\ &= \|\mathscr{H}\|_{1,1} \end{aligned} \tag{5.42}$$

and

$$\begin{aligned}\int_0^t \mathrm{d}\tau\, \mathscr{G}(\tau) &= \int_0^{p_1 t} \mathrm{d}\tau \frac{\mathscr{H}_1\left(\frac{\tau}{p_1}\right)}{p_1} + \int_{p_1 t}^{(p_1+p_2)t} \mathrm{d}\tau \frac{\mathscr{H}_2\left(\frac{\tau - p_1 t}{p_2}\right)}{p_2} \\ &\quad + \cdots + \int_{(p_1+p_2+\cdots+p_{\Gamma-1})t}^{t} \mathrm{d}\tau \frac{\mathscr{H}_\Gamma\left(\frac{\tau-(p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right)}{p_\Gamma} \\ &= \sum_{\gamma=1}^{\Gamma} \int_0^t \mathrm{d}\tau\, \mathscr{H}_\gamma(\tau).\end{aligned} \tag{5.43}$$

We use Statement 1 to deduce that

$$q(\tau) = \frac{\|\mathscr{G}(\tau)\|}{\|\mathscr{G}\|_1} = \begin{cases} \frac{\left\|\mathscr{H}_1\left(\frac{\tau}{p_1}\right)\right\|}{p_1 \|\mathscr{H}\|_{1,1}}, & 0 \le \tau < p_1 t, \\ \frac{\left\|\mathscr{H}_2\left(\frac{\tau - p_1 t}{p_2}\right)\right\|}{p_2 \|\mathscr{H}\|_{1,1}}, & p_1 t \le \tau < (p_1+p_2)t, \\ \quad \vdots & \\ \frac{\left\|\mathscr{H}_\Gamma\left(\frac{\tau-(p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right)\right\|}{p_\Gamma \|\mathscr{H}\|_{1,1}}, & (p_1+p_2+\cdots+p_{\Gamma-1})t \le \tau \le t. \end{cases} \tag{5.44}$$

Therefore,

$$
\begin{aligned}
&\int_0^t \mathrm{d}\tau\, q(\tau) e^{-i\frac{\mathscr{G}(\tau)}{q(\tau)}} \rho e^{i\frac{\mathscr{G}(\tau)}{q(\tau)}} \\
&= \int_0^{p_1 t} \mathrm{d}\tau \frac{\left\| \mathscr{H}_1\left(\frac{\tau}{p_1}\right)\right\|}{p_1 \left\|\mathscr{H}\right\|_{1,1}} \exp\left(- i \frac{\mathscr{H}_1\left(\frac{\tau}{p_1}\right)}{\left\| \mathscr{H}_1\left(\frac{\tau}{p_1}\right)\right\|} \left\|\mathscr{H}\right\|_{1,1} \right) \rho \exp\left(i \frac{\mathscr{H}_1\left(\frac{\tau}{p_1}\right)}{\left\| \mathscr{H}_1\left(\frac{\tau}{p_1}\right)\right\|} \left\|\mathscr{H}\right\|_{1,1} \right) \\
&+ \int_{p_1 t}^{(p_1+p_2)t} \mathrm{d}\tau \frac{\left\| \mathscr{H}_2\left(\frac{\tau-p_1 t}{p_2}\right)\right\|}{p_2 \left\|\mathscr{H}\right\|_{1,1}} \exp\left(- i \frac{\mathscr{H}_2\left(\frac{\tau-p_1 t}{p_2}\right)}{\left\| \mathscr{H}_2\left(\frac{\tau-p_1 t}{p_2}\right)\right\|} \left\|\mathscr{H}\right\|_{1,1} \right) \rho \exp\left(i \frac{\mathscr{H}_2\left(\frac{\tau-p_1 t}{p_2}\right)}{\left\| \mathscr{H}_2\left(\frac{\tau-p_1 t}{p_2}\right)\right\|} \left\|\mathscr{H}\right\|_{1,1} \right) \\
&+ \cdots + \int_{(p_1+p_2+\cdots+p_{\Gamma-1})t}^{t} \mathrm{d}\tau \frac{\left\| \mathscr{H}_\Gamma\left(\frac{\tau-(p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right)\right\|}{p_\Gamma \left\|\mathscr{H}\right\|_{1,1}} \\
&\cdot \exp\left(- i \frac{\mathscr{H}_\Gamma\left(\frac{\tau-(p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right)}{\left\| \mathscr{H}_\Gamma\left(\frac{\tau-(p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right)\right\|} \left\|\mathscr{H}\right\|_{1,1} \right) \rho \exp\left(i \frac{\mathscr{H}_\Gamma\left(\frac{\tau-(p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right)}{\left\| \mathscr{H}_\Gamma\left(\frac{\tau-(p_1+p_2+\cdots+p_{\Gamma-1})t}{p_\Gamma}\right)\right\|} \left\|\mathscr{H}\right\|_{1,1} \right) \\
&= \sum_{\gamma=1}^{\Gamma} \int_0^t \mathrm{d}\tau\, p_\gamma(\tau) e^{-i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}} \rho e^{i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}}, \qquad (5.45)
\end{aligned}
$$

which completes the proof of Statement 3. □

Theorem 29′ (L^1-norm error bound for continuous qDRIFT (LC), short-time version)**.** *Let $\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \mathscr{H}_\gamma(\tau)$ be a time-dependent Hamiltonian defined for $0 \leq \tau \leq t$ that is nonzero everywhere. Assume that each $\mathscr{H}_\gamma(\tau)$ is continuous and nonzero everywhere. Define $\mathrm{E}(t,0) = \exp_{\mathcal{T}}\big(- i \int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau)\big)$ and let $\mathcal{E}(t,0)(\cdot) = \mathrm{E}(t,0)(\cdot)\mathrm{E}^\dagger(t,0)$ be the corresponding quantum channel. Let $\mathcal{U}(t,0)$ be the continuous qDRIFT channel*

$$
\mathcal{U}(t,0)(\rho) := \sum_{\gamma=1}^{\Gamma} \int_0^t \mathrm{d}\tau\, p_\gamma(\tau) e^{-i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}} \rho e^{i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}}, \qquad (5.46)
$$

where $p_\gamma(\tau)$ is the probability distribution $p_\gamma(\tau) := \left\|\mathscr{H}_\gamma(\tau)\right\| / \left\|\mathscr{H}\right\|_{1,1}$. Then,

$$
\left\|\mathcal{E}(t,0) - \mathcal{U}(t,0)\right\|_\diamond \leq 4 \left\|\mathscr{H}\right\|_{1,1}^2 . \qquad (5.47)
$$

Proof. Consider the channel

$$\mathcal{G}(t,0)(\rho) := \int_0^t \mathrm{d}\tau\, q(\tau) e^{-i\frac{\mathscr{G}(\tau)}{q(\tau)}} \rho e^{i\frac{\mathscr{G}(\tau)}{q(\tau)}}, \tag{5.48}$$

where $q(\tau) := \|\mathscr{G}(\tau)\| / \|\mathscr{G}\|_1$ and $\mathscr{G}(\tau)$ is defined by (5.40). By Statement 3 of Theorem 32, it suffices to bound $\|\mathcal{E}(t,0) - \mathcal{G}(t,0)\|_\diamond$.

Define two parametrized quantum channels

$$\mathcal{E}_s(t,0)(\rho) = \mathrm{E}_s(t,0)\rho\mathrm{E}_s^\dagger(t,0), \qquad \mathcal{G}_s(t,0)(\rho) = \int_0^t \mathrm{d}\tau\, q(\tau) e^{-is\frac{\mathscr{G}(\tau)}{q(\tau)}} \rho e^{is\frac{\mathscr{G}(\tau)}{q(\tau)}} \tag{5.49}$$

and observe that

$$\mathcal{E}_0(t,0)(\rho) = \rho, \quad \mathcal{E}_1(t,0)(\rho) = \mathcal{E}(t,0)(\rho), \quad \mathcal{G}_0(t,0)(\rho) = \rho, \quad \mathcal{G}_1(t,0)(\rho) = \mathcal{G}(t,0)(\rho). \tag{5.50}$$

For readability, we only consider the trace norm $\|\mathcal{E}_1(t,0)(\rho) - \mathcal{G}_1(t,0)(\rho)\|_{\mathrm{tr}}$, whose analysis can be easily adapted to bound $\|(\mathcal{E}_1(t,0)\otimes\mathbb{1})(\sigma) - (\mathcal{G}_1(t,0)\otimes\mathbb{1})(\sigma)\|_{\mathrm{tr}}$ and thus the diamond-norm distance $\|\mathcal{E}_1(t,0) - \mathcal{G}_1(t,0)\|_\diamond$.

By Lemma 30 and Statement 2 of Theorem 32, we find that the first derivatives of $\mathcal{E}_s(t,0)(\rho)$ and $\mathcal{G}_s(t,0)(\rho)$ at $s=0$ agree with each other:

$$\frac{\mathrm{d}}{\mathrm{d}s}\mathcal{E}_s(t,0)(\rho)\Big|_{s=0} = \Big[-i\int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau), \rho\Big] = \Big[-i\int_0^t \mathrm{d}\tau\, \mathscr{G}(\tau), \rho\Big] = \frac{\mathrm{d}}{\mathrm{d}s}\mathcal{G}_s(t,0)(\rho)\Big|_{s=0}. \tag{5.51}$$

Thus, we can apply the fundamental theorem of calculus twice and obtain

$$
\begin{aligned}
&\mathcal{E}_1(t,0)(\rho) - \mathcal{G}_1(t,0)(\rho) \\
=&\big(\mathcal{E}_1(t,0)(\rho) - \mathcal{E}_0(t,0)(\rho)\big) - \big(\mathcal{G}_1(t,0)(\rho) - \mathcal{G}_0(t,0)(\rho)\big) \\
=&\int_0^1 \mathrm{d}s \int_0^s \mathrm{d}v \; \frac{\mathrm{d}^2}{\mathrm{d}v^2}\big[\mathcal{E}_v(t,0)(\rho) - \mathcal{G}_v(t,0)(\rho)\big] \\
=&\int_0^1 \mathrm{d}s \int_0^s \mathrm{d}v \; \Bigg\{ \frac{\mathrm{d}^2}{\mathrm{d}v^2}\mathrm{E}_v(t,0)\cdot\rho\cdot\mathrm{E}_v^\dagger(t,0) \\
&\qquad + 2\frac{\mathrm{d}}{\mathrm{d}v}\mathrm{E}_v(t,0)\cdot\rho\cdot\frac{\mathrm{d}}{\mathrm{d}v}\mathrm{E}_v^\dagger(t,0) + \mathrm{E}_v(t,0)\cdot\rho\cdot\frac{\mathrm{d}^2}{\mathrm{d}v^2}\mathrm{E}_v^\dagger(t,0) \\
&\qquad - \int_0^t \mathrm{d}\tau\, q(\tau) e^{-iv\frac{\mathscr{G}(\tau)}{q(\tau)}}\Big[-i\frac{\mathscr{G}(\tau)}{q(\tau)}, \Big[-i\frac{\mathscr{G}(\tau)}{q(\tau)}, \rho\Big]\Big] e^{iv\frac{\mathscr{G}(\tau)}{q(\tau)}} \Bigg\},
\end{aligned}
\tag{5.52}
$$

which implies

$$
\begin{aligned}
\left\|\mathcal{E}_1(t,0)(\rho) - \mathcal{G}_1(t,0)(\rho)\right\|_{\mathrm{tr}} &\leq \int_0^1 \mathrm{d}s \int_0^s \mathrm{d}v \; \Big\{ 2\left\|\mathscr{H}\right\|_{1,1}^2 + 2\left\|\mathscr{H}\right\|_{1,1}^2 + 4\left\|\mathscr{G}\right\|_1^2 \Big\} \\
&= 4\left\|\mathscr{H}\right\|_{1,1}^2 .
\end{aligned}
\tag{5.53}
$$

□

Theorem 31′ (L^1-norm error bound for continuous qDRIFT (LC), long-time version)**.** *Let* $\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \mathscr{H}_\gamma(\tau)$ *be a time-dependent Hamiltonian defined for* $0 \leq \tau \leq t$ *that is nonzero everywhere. Assume that each* $\mathscr{H}_\gamma(\tau)$ *is continuous and nonzero everywhere. Define* $\mathrm{E}(t,0) = \exp_{\mathcal{T}}\big(-i\int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau)\big)$ *and let* $\mathcal{E}(t,0)(\cdot) = \mathrm{E}(t,0)(\cdot)\mathrm{E}^\dagger(t,0)$ *be the corresponding quantum channel. Let* $\mathcal{U}(t,0)$ *be*

the continuous qDRIFT channel

$$\mathcal{U}(t,0)(\rho) := \sum_{\gamma=1}^{\Gamma} \int_0^t \mathrm{d}\tau \, p_\gamma(\tau) e^{-i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}} \rho e^{i\frac{\mathscr{H}_\gamma(\tau)}{p_\gamma(\tau)}}, \tag{5.54}$$

where $p_\gamma(\tau)$ *is the probability distribution* $p_\gamma(\tau) := \|\mathscr{H}_\gamma(\tau)\| / \|\mathscr{H}\|_{1,1}$. *Then, for any positive integer* r, *there exists a division* $0 = t_0 < t_1 < \cdots < t_r = t$, *such that*

$$\left\| \mathcal{E}(t,0) - \prod_{j=0}^{r-1} \mathcal{U}(t_{j+1}, t_j) \right\|_\diamond \leq 4 \frac{\|\mathscr{H}\|_{1,1}^2}{r}. \tag{5.55}$$

To ensure that the simulation error is at most ϵ, *it thus suffices to choose*

$$r \geq 4 \left\lceil \frac{\|\mathscr{H}\|_{1,1}^2}{\epsilon} \right\rceil. \tag{5.56}$$

The proof of this theorem follows from Theorem 29′ using the same reasoning as that used to prove Theorem 31.

5.4 Discussion

We have shown that a time-dependent Hamiltonian $\mathscr{H}(\tau)$ can be simulated for time $0 \leq \tau \leq t$ with gate complexity that scales according to the L^1 norm $\int_0^t \mathrm{d}\tau \, \|\mathscr{H}(\tau)\|$. We achieve this by developing a new simulation algorithm based on classical sampling. Although we have assumed that the input Hamiltonians are given in the LC model, our analysis can be extended to the simulation of sparse

Hamiltonians. The idea is to use a time-dependent version of Lemma 6 to represet the input sparse Hamiltonian as a linear combination of operators [16, Section 3.3]. In both cases, this result is a polynomial speedup in terms of the norm dependence, an advantage that can be favorable in practice. In particular, this can potentially be applied to simulating scattering processes in quantum chemistry [16, Section 5].

Besides the randomization approach, we can also use a rescaling principle for the Schrödinger equation to improve time-dependent Hamiltonian simulation. In the rescaled Schrödinger equation, the time-dependent Hamiltonian $\mathscr{H}(\tau)$ has the same norm at all $\tau \in [0, t]$, so the norm inequality $\int_0^t \mathrm{d}\tau \, \|\mathscr{H}(\tau)\| \leq t \max_{\tau \in [0,t]} \|\mathscr{H}(\tau)\|$ holds with equality. Using this principle, it is possible to show that the simulation algorithm based on the truncated Dyson series [14, 65, 76] can also be improved to have L^1-norm scaling. Further discussion of this approach is beyond the scope of this dissertation, and we refer the reader to [16, Section 4] for details.

For most of our analysis, we have assumed that the Hamiltonian $\mathscr{H}(\tau)$ is continuous. This assumption can be relaxed to allow finitely many discontinuities. In fact, the continuous qDRIFT algorithm works properly provided only that $\mathscr{H}(\tau)$ is Lebesgue integrable (see [42] for details). Our analysis can also be adapted to simulate time-dependent Hamiltonians that have countably many zeros. Indeed, since the equation $\mathscr{H}(\tau) = 0$ has at most countably many solutions, we can find

$c \in \mathbb{R}$ such that $\mathscr{H}(\tau) + cI$ is nonzero everywhere. Then, $\exp_{\mathcal{T}}\big(-i \int_0^t \mathrm{d}\tau\, (\mathscr{H}(\tau) + cI)\big) = e^{-ict} \exp_{\mathcal{T}}\big(-i \int_0^t \mathrm{d}\tau\, \mathscr{H}(\tau)\big)$, so the result is only off by a global phase. Note that this assumption can be completely dropped if we use continuous qDRIFT: we define the exceptional set

$$\mathcal{B}_0 := p^{-1}(0) = \{\tau : p(\tau) = 0\} = \{\tau : \|\mathscr{H}(\tau)\| = 0\} = \{\tau : \mathscr{H}(\tau) = 0\} \tag{5.57}$$

and redefine $\mathcal{U}(t, 0)$ as

$$\mathcal{U}(t,0)(\rho) := \int_{[0,t]\setminus\mathcal{B}_0} \mathrm{d}\tau\; p(\tau) e^{-i\frac{\mathscr{H}(\tau)}{p(\tau)}} \rho e^{i\frac{\mathscr{H}(\tau)}{p(\tau)}}, \qquad p(\tau) := \frac{\|\mathscr{H}(\tau)\|}{\|\mathscr{H}\|_1}. \tag{5.58}$$

We note that $\mathcal{U}(t, 0)$ is a valid quantum channel and can be implemented with unit cost. Indeed, for any input state ρ, we randomly sample a value τ according to $p(\tau)$ and perform $e^{-i\mathscr{H}(\tau)/p(\tau)}$ if $\tau \in [0, t]\setminus\mathcal{B}_0$, and the identity operation otherwise. This implements

$$\int_{[0,t]\setminus\mathcal{B}_0} \mathrm{d}\tau\; p(\tau) e^{-i\frac{\mathscr{H}(\tau)}{p(\tau)}} \rho e^{i\frac{\mathscr{H}(\tau)}{p(\tau)}} + \int_{\mathcal{B}_0} \mathrm{d}\tau\; p(\tau)\rho = \mathcal{U}(t,0)(\rho). \tag{5.59}$$

The remaining analysis proceeds as in Section 5.2 and Section 5.3.

The qDRIFT protocol that we analyzed here only achieves first-order accuracy. It is natural to ask if sampling a different probability distribution could lead to an algorithm with better performance. The answer seems to be "no" if

we are restricted to a univariate distribution. To see this, consider the discrete case where $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ is a Hamiltonian consisting of Γ terms. We sample according to a probability vector $p \in [0,1]^\Gamma$. Upon getting outcome γ, we perform the unitary e^{-itH_γ/p_γ}. Effectively, we implement the quantum channel $\mathcal{U}(t)(\rho) := \sum_{\gamma=1}^{\Gamma} p_\gamma e^{-it\frac{H_\gamma}{p_\gamma}} \rho e^{it\frac{H_\gamma}{p_\gamma}}$, which is a first-order approximation to the ideal evolution $\mathcal{E}(t)(\rho) := e^{-it\sum_{\gamma=1}^{\Gamma} H_\gamma} \rho e^{it\sum_{\gamma=1}^{\Gamma} H_\gamma}$. In particular, the difference between $\mathcal{U}(t)(\rho)$ and $\mathcal{E}(t)(\rho)$ admits an integral representation

$$\begin{aligned}
&\mathcal{U}(t)(\rho) - \mathcal{E}(t)(\rho) \\
&= \int_0^t \mathrm{d}u \int_0^u \mathrm{d}v \ \left\{ \sum_{\gamma=1}^{\Gamma} p_\gamma e^{-iv\frac{H_\gamma}{p_\gamma}} \left[-i\frac{H_\gamma}{p_\gamma}, \left[-i\frac{H_\gamma}{p_\gamma}, \rho \right] \right] e^{iv\frac{H_\gamma}{p_\gamma}} \right. \\
&\qquad \left. - e^{-iv\sum_{\gamma=1}^{\Gamma} H_\gamma} \left[-i\sum_{\gamma=1}^{\Gamma} H_\gamma, \left[-i\sum_{\gamma=1}^{\Gamma} H_\gamma, \rho \right] \right] e^{iv\sum_{\gamma=1}^{\Gamma} H_\gamma} \right\}.
\end{aligned} \tag{5.60}$$

To estimate the diamond-norm error $\|\mathcal{U}(t) - \mathcal{E}(t)\|_\diamond$, we take σ to be a state on the joint system of the original register and an ancilla register with the same dimension. We compute

$$\begin{aligned}
&\|(\mathcal{U}(t) \otimes \mathbb{1})(\sigma) - (\mathcal{E}(t) \otimes \mathbb{1})(\sigma)\|_{\mathrm{tr}} \\
&\leq \int_0^t \mathrm{d}u \int_0^u \mathrm{d}v \ \left\{ \sum_{\gamma=1}^{\Gamma} p_\gamma \left\| \left[-i\frac{H_\gamma}{p_\gamma} \otimes \mathbb{1}, \left[-i\frac{H_\gamma}{p_\gamma} \otimes \mathbb{1}, \sigma \right] \right] \right\|_{\mathrm{tr}} \right. \\
&\qquad \left. + \left\| \left[-i\sum_{\gamma=1}^{\Gamma} H_\gamma \otimes \mathbb{1}, \left[-i\sum_{\gamma=1}^{\Gamma} H_\gamma \otimes \mathbb{1}, \sigma \right] \right] \right\|_{\mathrm{tr}} \right\} \\
&\leq 2t^2 \left(\sum_{\gamma=1}^{\Gamma} \frac{\|H_\gamma\|^2}{p_\gamma} + \|H\|_1^2 \right).
\end{aligned} \tag{5.61}$$

By Jensen's inequality,

$$\sum_{\gamma=1}^{\Gamma} \frac{\|H_\gamma\|^2}{p_\gamma} = \sum_{\gamma=1}^{\Gamma} p_\gamma \left(\frac{\|H_\gamma\|}{p_\gamma} \right)^2 \geq \left(\sum_{\gamma=1}^{\Gamma} p_\gamma \frac{\|H_\gamma\|}{p_\gamma} \right)^2 = \|H\|_1^2 \,, \tag{5.62}$$

with equality if and only if all $\|H_\gamma\|/p_\gamma$ are equal, implying that the probability distribution $p_\gamma := \|H_\gamma\|/\|H\|_1$ is optimal. A similar optimality result holds for continuous qDRIFT (though the proof is more involved).

However, this does not preclude the existence of a higher-order qDRIFT protocol using more complicated sampling. For example, besides the basic evolutions e^{-itH_γ/p_γ}, one could evolve under commutators $[H_j, H_k]$ or anticommutators $\{H_j, H_k\}$. We could also use a multivariate distribution and correlate different steps of the qDRIFT protocol. For future work, it would be interesting to find a higher-order protocol, or prove that such a protocol cannot exist.

Finally, it would be interesting to identify concrete algorithmic applications of Hamiltonian simulation with L^1-norm scaling. It might also be of interest to demonstrate these approaches experimentally, for applications such as implementing adiabatic algorithms with quantum circuits.

Chapter 6: Analysis of product formulas: general theory

We have seen in Chapter 3 that there exists a significant gap between the provable and actual performance of product formulas. In this chapter, we address this by developing a general theory for analyzing the error of product formulas (Trotter error). We summarize prior approaches to analyzing Trotter error and discuss their limitations in Section 6.1. We then present a new Trotter error analysis. Specifically, we consider various types of Trotter error in Section 6.3 and derive their order conditions in Section 6.4. We then develop a representation of Trotter error in Section 6.5 that directly exploits the commutativity of the simulated system. We illustrate these ideas in Section 6.2 with the simple example of the first-order Lie-Trotter formula.

In our derivation, we work in a general setting where the input operators are not necessarily Hermitian/anti-Hermitian. This allows us to simultaneously handle real-time evolutions for digital quantum simulation (Chapter 7) and imaginary-time evolutions for quantum Monte Carlo simulation (Chapter 9).

This chapter is partly based on the following paper:

[37] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu, *A theory of Trotter error*, 2019, arXiv:1912.08854.

6.1 Previous analyses of Trotter error

We now briefly summarize prior approaches to analyzing Trotter error and discuss their limitations.

The original work of Lloyd [70] analyzes product formulas by truncating the Taylor expansion (or the BCH expansion). Recall that the Lie-Trotter formula $\mathscr{S}_1(t)$ provides a first-order approximation to the evolution, so $\mathscr{S}_1(t) = e^{-itH} + O(t^2)$. To better analyze the Trotter error, Lloyd dropped all higher-order terms in the Taylor expansion and focused only on the terms of lowest order t^2. This approach is intuitive and has been employed by subsequent works to give rough estimation of Trotter error. The drawback of this analysis is that it implicitly assumes that the high-order terms are dominated by the lowest-order term. However, this does not necessarily hold for many systems such as nearest-neighbor lattice Hamiltonians [36] and chemical Hamiltonians [107] when the time step t is fixed.

This issue was addressed in the seminal work of Berry, Ahokas, Cleve, and Sanders by using a tail bound of the Taylor expansion [12]. This gave, for the first time, a concrete bound on the Trotter error for high-order Suzuki formulas. For a Hamiltonian $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ containing Γ summands, their bound scales with $\Gamma \max_\gamma \|H_\gamma\|$, although it is not hard to improve this [54] to $\sum_{\gamma=1}^{\Gamma} \|H_\gamma\|$ [75, 99]. Regardless of which scaling to use, this worst-case analysis does not exploit the

commutativity of Hamiltonian summands and the resulting complexity is worse than many post-Trotter methods.

Error bounds that exploit the commutativity of summands are known for low-order formulas, such as the Lie-Trotter formula [61, 99] and the second-order Suzuki formula [41, 66, 99, 107]. These bounds are tight in the sense that they match the lowest-order term of the BCH expansion up to an application of the triangle inequality. However, it is unclear whether they can be generalized, say, to the fourth- or the sixth-order case, which are still reasonably simple and can provide a significant advantage in practice [34].

Instead, previous works made compromises to obtain improved analyses of higher-order formulas. Somma gave an improved bound by representing the Trotter error as an infinite series of nested commutators [98]. This approach is advantageous when the simulated system has an underlying Lie-algebraic structure with small structure factors, such as for a quantum harmonic oscillator and certain nonquadratic potentials. However, this reduces to the worst-case analysis of Berry, Ahokas, Cleve, and Sanders for other systems. An alternative approach suggested by Childs et al. exploited commutativity of the lowest-order error terms and estimated higher-order ones using a tail bound for the Taylor series [34]. This analysis is bottlenecked by the tail bound, so it only offers a modest improvement over the worst-case analysis.

We will give a new bound on the Trotter error in Theorem 39 that depends

on nested commutators of the operator summands, overcoming the limitations of all prior error analyses of product formulas.

6.2 Example of the Lie-Trotter formula

In this section, we use the example of the first-order Lie-Trotter formula to illustrate the general theory we develop for analyzing Trotter error. For simplicity, consider an operator $H = A + B$ with two summands. The ideal evolution generated by H is given by $e^{tH} = e^{t(A+B)}$. To decompose this evolution, we may use the Lie-Trotter formula $\mathscr{S}_1(t) = e^{tB}e^{tA}$. This formula is first-order accurate, so we have $\mathscr{S}_1(t) = e^{tH} + O(t^2)$.

A key observation here is that the error of a product formula can have various *types*. Specifically, we consider three types of Trotter error: additive error, multiplicative error, and error that appears in the exponent. Note that $\mathscr{S}_1(t)$ satisfies the differential equation $\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}_1(t) = H\mathscr{S}_1(t) + \big[e^{tB}, A\big]e^{tA}$ with initial condition $\mathscr{S}_1(0) = I$. By the variation-of-parameters formula (Lemma 1),

$$\mathscr{S}_1(t) = e^{tH} + \int_0^t \mathrm{d}\tau \; e^{(t-\tau)H}\big[e^{\tau B}, A\big]e^{\tau A}, \tag{6.1}$$

so we get the additive error $\mathscr{A}_1(t) = \int_0^t \mathrm{d}\tau \; e^{(t-\tau)H}\big[e^{\tau B}, A\big]e^{\tau A}$ of the Lie-Trotter formula. For error with the exponentiated type, we differentiate $\mathscr{S}_1(t)$ to get $\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}_1(t) = \big(B + e^{tB}Ae^{-tB}\big)\mathscr{S}_1(t)$. Applying the fundamental theorem of time-

ordered evolution (Lemma 3), we have

$$\mathscr{S}_1(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\big(B + e^{\tau B} A e^{-\tau B}\big)\right), \tag{6.2}$$

and so $\mathscr{E}_1(\tau) = e^{\tau B} A e^{-\tau B} - A$ is the error of Lie-Trotter formula that appears in the exponent. To obtain the multiplicative error, we switch to the interaction picture using Lemma 2:

$$\mathscr{S}_1(t) = e^{tH} \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\big(e^{-\tau H} e^{\tau B} A e^{-\tau B} e^{\tau H} - e^{-\tau H} A e^{\tau H}\big)\right), \tag{6.3}$$

so $\mathscr{M}_1(t) = \exp_{\mathcal{T}}\big(\int_0^t \mathrm{d}\tau\big(e^{-\tau H} e^{\tau B} A e^{-\tau B} e^{\tau H} - e^{-\tau H} A e^{\tau H}\big)\big) - I$ is the multiplicative Trotter error. These three types of Trotter error are equivalent for analyzing the complexity of digital quantum simulation (Chapter 7), whereas the multiplicative error and the exponentiated error are more versatile when applied to quantum Monte Carlo simulation (Chapter 9). We compute error operators for a general product formula in Section 6.3.

Since product formulas provide a good approximation to the ideal evolution for small t, we expect all three error operators $\mathscr{A}_1(t)$, $\mathscr{E}_1(t)$, and $\mathscr{M}_1(t)$ to converge to zero in the limit $t \to 0$. The rates of convergence are what we call *order*

conditions. More precisely,

$$\begin{aligned}
\mathscr{A}_1(t) &= \int_0^t \mathrm{d}\tau \; e^{(t-\tau)H}\big[e^{\tau B}, A\big]e^{\tau A} = O\left(t^2\right), \\
\mathscr{E}_1(t) &= e^{tB}Ae^{-tB} - A = O\left(t\right), \\
\mathscr{M}_1(t) &= \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\big(e^{-\tau H}e^{\tau B}Ae^{-\tau B}e^{\tau H} - e^{-\tau H}Ae^{\tau H}\big)\right) - I = O\left(t^2\right).
\end{aligned} \tag{6.4}$$

For the Lie-Trotter formula, these conditions can be verified by direct calculation, although such an approach becomes inefficient in general. Instead, we describe an indirect approach in Section 6.4 to compute order conditions for a general product formula.

Finally, we consider *representations* of Trotter error that leverage the commutativity of operator summands. We discuss how to represent $\mathscr{M}_1(t)$ in detail, although it is straightforward to extend the analysis to $\mathscr{A}_1(t)$ and $\mathscr{E}_1(t)$ as well. To this end, we first consider the term $e^{-\tau H}e^{\tau B}Ae^{-\tau B}e^{\tau H}$, which contains two layers of conjugations of matrix exponentials. We apply the fundamental theorem of calculus to the first layer of conjugation and obtain

$$e^{\tau B}Ae^{-\tau B} = A + \int_0^\tau \mathrm{d}\tau_2 \; e^{\tau_2 B}\big[B, A\big]e^{-\tau_2 B}. \tag{6.5}$$

After cancellation, this gives

$$\mathscr{M}_1(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \int_0^\tau \mathrm{d}\tau_2 \; e^{-\tau H}e^{\tau_2 B}\big[B, A\big]e^{-\tau_2 B}e^{\tau H}\right) - I, \tag{6.6}$$

which implies, through Corollary 5, that $\|\mathscr{M}_1(t)\| = \mathcal{O}\left(\|[B,A]\|\, t^2\right)$ when A, B are anti-Hermitian and $t \geq 0$, and that $\|\mathscr{M}_1(t)\| = \mathcal{O}\left(\|[B,A]\|\, t^2 e^{2t(\|A\|+\|B\|)}\right)$ in general. In the above derivation, it is important that we only expand the first layer of conjugation of exponentials, that we apply the fundamental theorem of calculus only once, and that we can cancel the terms $e^{-\tau H} A e^{\tau H}$ in pairs. The validity of such an approach in general is guaranteed by the appropriate order condition, which we explain in detail in Section 6.5.

6.3 Error types

In this section, we discuss error types of a general product formula. In particular, we give explicit expressions for three different types of Trotter error: the additive error, the multiplicative error, and error that appears in the exponent of a time-ordered exponential (the "exponentiated" error). These types are equivalent for analyzing the complexity of simulating quantum dynamics and local observables, but the latter two types are more versatile for quantum Monte Carlo simulation.

Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be an operator with Γ summands. The ideal evolution under H for time t is given by $e^{tH} = e^{t\sum_{\gamma=1}^{\Gamma} H_\gamma}$, which we approximate by a general product formula $\mathscr{S}(t) = \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}$. For convenience, we use the lexicographic order on a pair of tuples (υ,γ) and (υ',γ'), defined as follows: we write $(\upsilon,\gamma) \succeq (\upsilon',\gamma')$ if $\upsilon > \upsilon'$, or if $\upsilon = \upsilon'$ and $\gamma \geq \gamma'$. We have $(\upsilon,\gamma) \succ (\upsilon',\gamma')$

if both $(\upsilon,\gamma) \succeq (\upsilon',\gamma')$ and $(\upsilon,\gamma) \neq (\upsilon',\gamma')$ hold. Notations $(\upsilon,\gamma) \preceq (\upsilon',\gamma')$ and $(\upsilon,\gamma) \prec (\upsilon',\gamma')$ are defined in a similar way, except that we reverse the directions of all the inequalities.

To compute the additive error, we construct the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}(t) = H\mathscr{S}(t) + \mathscr{R}(t), \tag{6.7}$$

with initial condition $\mathscr{S}(0) = I$, where

$$\begin{aligned}\mathscr{R}(t) := \sum_{(\upsilon,\gamma)} \overleftarrow{\prod_{(\upsilon',\gamma')\succ(\upsilon,\gamma)}} e^{ta_{(\upsilon',\gamma')}H_{\pi_{\upsilon'}(\gamma')}} \left(a_{(\upsilon,\gamma)}H_{\pi_\upsilon(\gamma)}\right) \overleftarrow{\prod_{(\upsilon',\gamma')\preceq(\upsilon,\gamma)}} e^{ta_{(\upsilon',\gamma')}H_{\pi_{\upsilon'}(\gamma')}} \\ - H \overleftarrow{\prod_{(\upsilon',\gamma')}} e^{ta_{(\upsilon',\gamma')}H_{\pi_{\upsilon'}(\gamma')}}.\end{aligned} \tag{6.8}$$

By the variation-of-parameters formula (Lemma 1), $\mathscr{S}(t) - e^{tH} = \int_0^t \mathrm{d}\tau\, e^{(t-\tau)H}\mathscr{R}(\tau)$, so we obtain the additive error

$$\mathscr{A}(t) := \int_0^t \mathrm{d}\tau\, e^{(t-\tau)H}\mathscr{R}(\tau). \tag{6.9}$$

This suffices if our purpose is to only compute the additive error operator. However, for the later discussion in Section 6.5, it is convenient to further rewrite

$$\mathscr{A}(t) = \int_0^t \mathrm{d}\tau\, e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau), \tag{6.10}$$

where

$$\begin{aligned}\mathscr{T}(\tau) := &\sum_{(\upsilon,\gamma)} \overrightarrow{\prod_{(\upsilon',\gamma')\prec(\upsilon,\gamma)}} e^{-\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \left(a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}\right) \overleftarrow{\prod_{(\upsilon',\gamma')\prec(\upsilon,\gamma)}} e^{\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \\ &- \overrightarrow{\prod_{(\upsilon',\gamma')}} e^{-\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} H \overleftarrow{\prod_{(\upsilon',\gamma')}} e^{\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}}.\end{aligned} \tag{6.11}$$

Note that we have rewritten part of the error operator as a linear combination of conjugation of matrix exponentials. In Section 6.5, we apply the correct order condition to further represent it as nested commutators of the operator summands H_γ.

For the exponentiated type of Trotter error, we aim to construct an operator-valued function $\mathscr{E}(t)$ such that

$$\mathscr{S}(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \big(H + \mathscr{E}(\tau)\big)\right). \tag{6.12}$$

To do this, we differentiate the product formula $\mathscr{S}(t)$ and obtain

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}(t) &= \sum_{(\upsilon,\gamma)} \overleftarrow{\prod_{(\upsilon',\gamma')\succ(\upsilon,\gamma)}} e^{t a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \left(a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}\right) \overleftarrow{\prod_{(\upsilon',\gamma')\preceq(\upsilon,\gamma)}} e^{t a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \\ &= \mathscr{F}(t)\mathscr{S}(t),\end{aligned} \tag{6.13}$$

where

$$\mathscr{F}(t) := \sum_{(v,\gamma)} \overleftarrow{\prod_{(v',\gamma')\succ(v,\gamma)}} e^{t a_{(v',\gamma')} H_{\pi_{v'}(\gamma')}} \left(a_{(v,\gamma)} H_{\pi_v(\gamma)}\right) \overrightarrow{\prod_{(v',\gamma')\succ(v,\gamma)}} e^{-t a_{(v',\gamma')} H_{\pi_{v'}(\gamma')}}. \tag{6.14}$$

Applying the fundamental theorem of time-ordered evolution (Lemma 3), we have

$$\mathscr{S}(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{F}(\tau)\right), \tag{6.15}$$

which gives the exponentiated error

$$\mathscr{E}(t) := \mathscr{F}(t) - H. \tag{6.16}$$

From the exponentiated type of Trotter error, we can obtain the multiplicative error by switching to the interaction picture. Specifically, we apply Lemma 2 and get

$$\mathscr{S}(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\big(H + \mathscr{E}(\tau)\big)\right) = e^{tH} \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ e^{-\tau H}\mathscr{E}(\tau)e^{\tau H}\right). \tag{6.17}$$

Then, the operator-valued function

$$\mathscr{M}(t) := \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ e^{-\tau H}\mathscr{E}(\tau)e^{\tau H}\right) - I \tag{6.18}$$

is the multiplicative error of the product formula. We have thus established:

Theorem 33 (Types of Trotter error)**.** *Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be an operator with Γ summands. The evolution under H for time $t \in \mathbb{R}$ is given by $e^{tH} = e^{t\sum_{\gamma=1}^{\Gamma} H_\gamma}$, which we decompose using the product formula $\mathscr{S}(t) = \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}$. Then,*

1. *Trotter error can be expressed in the additive form $\mathscr{S}(t) = e^{tH} + \int_0^t \mathrm{d}\tau\ e^{(t-\tau)H} \cdot \mathscr{S}(\tau)\mathscr{T}(\tau)$, where*

$$\mathscr{T}(\tau) = \sum_{(\upsilon,\gamma)} \overrightarrow{\prod_{(\upsilon',\gamma') \prec (\upsilon,\gamma)}} e^{-\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \left(a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}\right) \overleftarrow{\prod_{(\upsilon',\gamma') \prec (\upsilon,\gamma)}} e^{\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} - \overrightarrow{\prod_{(\upsilon',\gamma')}} e^{-\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} H \overleftarrow{\prod_{(\upsilon',\gamma')}} e^{\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}}; \tag{6.19}$$

2. *Trotter error can be expressed in the exponentiated form $\mathscr{S}(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \left(H + \mathscr{E}(\tau)\right)\right)$, where*

$$\mathscr{E}(\tau) = \sum_{(\upsilon,\gamma)} \overleftarrow{\prod_{(\upsilon',\gamma') \succ (\upsilon,\gamma)}} e^{\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \left(a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}\right) \overrightarrow{\prod_{(\upsilon',\gamma') \succ (\upsilon,\gamma)}} e^{-\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} - H; \tag{6.20}$$

3. *Trotter error can be expressed in the multiplicative form $\mathscr{S}(t) = e^{tH}(I + \mathscr{M}(t))$, where*

$$\mathscr{M}(t) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ e^{-\tau H} \mathscr{E}(\tau) e^{\tau H}\right) - I \tag{6.21}$$

with $\mathscr{E}(\tau)$ as above.

Note that the error operators $\mathscr{T}(\tau)$ and $\mathscr{E}(\tau)$ both consist of conjugations of matrix exponentials of the form $e^{\tau A_s} \cdots e^{\tau A_2} e^{\tau A_1} B e^{-\tau A_1} e^{-\tau A_2} \cdots e^{-\tau A_s}$. To bound the Trotter error, it thus suffices to analyze such conjugations of matrix exponentials. The previous work of Somma [98] expanded them into infinite series of nested commutators, which is favorable for systems with appropriate Lie-algebraic structures. An alternative approach of Childs and Su [36] represented them as commutators nested with conjugations of matrix exponentials, which provides a tight analysis for geometrically local systems. Unfortunately, both approaches can be loose in general. Instead, we apply order conditions (Section 6.4) and derive a new representation of Trotter error (Section 6.5) that provides a tight analysis for general systems.

6.4 Order conditions

In this section, we study the order conditions of Trotter error. By order condition, we mean the rate at which a continuous operator-valued function $\mathscr{F}(\tau)$, defined for $\tau \in \mathbb{R}$, approaches zero in the limit $\tau \to 0$. Formally, we write $\mathscr{F}(\tau) = O(\tau^p)$ with nonnegative integer p if there exist constants $c, t_0 > 0$, independent of τ, such that $\|\mathscr{F}(\tau)\| \leq c\,|\tau|^p$ whenever $|\tau| \leq t_0$.

Order conditions arise naturally in the analysis of Trotter error [4, 5, 100, 110]. Indeed, a pth-order product formula $\mathscr{S}(t)$ has a Taylor expansion that agrees with the ideal evolution e^{tH} up to order t^p. This implies the order condition

$\mathscr{S}(t) = e^{tH} + O\left(t^{p+1}\right)$ by definition. Our approach is to use this relation in the reverse direction: given a smooth operator-valued function $\mathscr{F}(\tau)$ satisfying the order condition $\mathscr{F}(\tau) = O\left(\tau^p\right)$, we conclude that $\mathscr{F}(\tau)$ has a Taylor expansion where terms with order τ^{p-1} or lower vanish.

Formally, given a continuous operator-valued function $\mathscr{F}(\tau)$ defined on $\mathbb{R}$, we write $\mathscr{F}(\tau) = O(\tau^p)$ with nonnegative integer p if there exist constants $c, t_0 > 0$, independent of τ, such that $\|\mathscr{F}(\tau)\| \leq c\,|\tau|^p$ whenever $|\tau| \leq t_0$. To verify this, it suffices to check that the limit

$$\lim_{\tau \to 0} \frac{\|\mathscr{F}(\tau)\|}{|\tau|^p} \tag{6.22}$$

exists.

As aforementioned, our approach uses the order condition $\mathscr{F}(\tau) = O\left(\tau^p\right)$ to argue that terms with order $1, \tau, \ldots, \tau^{p-1}$ vanish in the Taylor series of $\mathscr{F}(\tau)$. This argument is rigorized in [36, Lemma 6], which we restate and prove for completeness.

Lemma 34 (Derivative condition)**.** *Any continuous operator-valued function $\mathscr{F}(\tau)$ defined on $\mathbb{R}$ satisfies the order condition*

$$\mathscr{F}(\tau) = O\left(1\right). \tag{6.23}$$

Furthermore, if $\mathscr{F}(\tau)$ has p continuous derivatives for some positive integer p, then

the following two conditions are equivalent:

1. $\mathscr{F}(\tau) = O(\tau^p)$; *and*

2. $\mathscr{F}(0) = \mathscr{F}'(0) = \cdots = \mathscr{F}^{(p-1)}(0) = 0$.

Proof. The continuity of $\mathscr{F}(\tau)$ at $\tau = 0$ implies $\mathscr{F}(\tau) = O(1)$ by definition. Assume that $\mathscr{F}(\tau)$, $\mathscr{F}'(\tau)$,..., $\mathscr{F}^{(p)}(\tau)$ exists and are continuous. If Condition 2 holds, we have

$$\lim_{\tau\to 0} \frac{\|\mathscr{F}(\tau)\|}{|\tau|^p} = \left\| \lim_{\tau\to 0} \frac{\mathscr{F}(\tau)}{\tau^p} \right\| = \frac{\left\|\mathscr{F}^{(p)}(0)\right\|}{p!} \tag{6.24}$$

by the L'Hôpital's rule. This proves that Condition 1 holds.

Given Condition 1, we have by definition that

$$\|\mathscr{F}(\tau)\| \leq c\,|\tau|^p \tag{6.25}$$

for some $c, t_0 > 0$ and all $|\tau| \leq t_0$. Suppose by contradiction that Condition 2 is not true. Then we let $0 \leq j \leq p-1$ be the first integer for which $\mathscr{F}^{(j)}(0) \neq 0$. We use the Taylor's theorem to order j to get

$$\mathscr{F}(\tau) = \mathscr{F}^{(j)}(0)\frac{\tau^j}{j!} + \int_0^\tau \mathrm{d}\tau_2\ \mathscr{F}^{(j+1)}(\tau - \tau_2)\frac{\tau_2^j}{j!}, \tag{6.26}$$

which implies

$$\|\mathscr{F}(\tau)\| \geq \left\|\mathscr{F}^{(j)}(0)\right\| \frac{|\tau|^j}{j!} - \max_{|\tau_2|\leq|\tau|} \left\|\mathscr{F}^{(j+1)}(\tau_2)\right\| \frac{|\tau|^{j+1}}{(j+1)!} \tag{6.27}$$

by the triangle inequality. We combine the above inequalities and divide both sides by $|\tau|^j$. Taking the limit $\tau \to 0$ gives the contradiction $\left\|\mathscr{F}^{(j)}(0)\right\| \leq 0$. $\square$

We can determine the order condition of an operator-valued function through either direct calculation or indirect derivation. To illustrate this, we consider decomposing $e^{tH} = e^{t(A+B)}$ using the first-order Lie-Trotter formula $\mathscr{S}_1(t) = e^{tB}e^{tA}$. We see from Section 6.2 that this decomposition has the additive Trotter error

$$\mathscr{A}_1(t) = \int_0^t \mathrm{d}\tau \; e^{(t-\tau)H}\big(\mathscr{S}_1'(\tau) - H\mathscr{S}_1(\tau)\big) = \int_0^t \mathrm{d}\tau \; e^{(t-\tau)H}\big[e^{\tau B}, A\big]e^{\tau A}. \tag{6.28}$$

We know that $\mathscr{A}_1(t)$ has order condition $\mathscr{A}_1(t) = O\left(t^2\right)$, which follows directly from the fact that $\mathscr{A}_1(0) = \mathscr{A}_1'(0) = 0$. On the other hand, an indirect argument would proceed as follows. We use the known order condition $\mathscr{S}_1(t) = e^{tH} + O\left(t^2\right)$ to conclude that $\mathscr{S}_1'(\tau) - H\mathscr{S}_1(\tau) = O\left(\tau\right)$. Multiplying the matrix exponential $e^{(t-\tau)H} = O\left(1\right)$ does not change the order condition, so we still have $e^{(t-\tau)H}\big(\mathscr{S}_1'(\tau) - H\mathscr{S}_1(\tau)\big) = O\left(\tau\right)$. A final integration of $\int_0^t \mathrm{d}\tau$ then gives the desired condition $\mathscr{A}_1(t) = O\left(t^2\right)$.

Lemma 34 provides a direct approach to computing order conditions for functions of real variables. This works for simple examples such as the power functions $f(\tau) = \tau^p = O\left(\tau^p\right)$. Another example which we will use in our analysis

is the integration of a monomial, like

$$\int_0^{\tau} \mathrm{d}\tau_1 \int_0^{\tau_1} \mathrm{d}\tau_2 \int_0^{\tau_1} \mathrm{d}\tau_3 \int_0^{\tau_2} \mathrm{d}\tau_4 \; \tau_1^3 \tau_2 \tau_3^4 \tau_4^5. \tag{6.29}$$

As the following lemma shows, we can directly evaluate such an integral and compute the order condition of the resulting power function.

Lemma 35 (Integration of a monomial)**.** *The integration of a monomial* $\tau_1^{p_1} \cdots \tau_\gamma^{p_\gamma} \cdots \tau_\Gamma^{p_\Gamma}$ *evaluates as*

$$\int_0^{\tau} \mathrm{d}\tau_1 \cdots \int_0^{\tau_{<\gamma}} \mathrm{d}\tau_\gamma \cdots \int_0^{\tau_{<\Gamma}} \mathrm{d}\tau_\Gamma \; \tau_1^{p_1} \cdots \tau_\gamma^{p_\gamma} \cdots \tau_\Gamma^{p_\Gamma} = c t^{p_1 + \cdots + p_\Gamma + \Gamma} = O\left(t^{p_1 + \cdots + p_\Gamma + \Gamma}\right), \tag{6.30}$$

where $\tau_{<\gamma} \in \{\tau, \tau_1, \ldots, \tau_{\gamma-1}\}$ *and* c *is a constant that depends on nonnegative integers* $p_1, \ldots, p_\Gamma$*.*

Proof. We induct on the value of Γ. The claim trivially holds when $\Gamma = 1$. Suppose that it is true for Γ. For $\Gamma + 1$, we have

$$\int_0^{\tau} \mathrm{d}\tau_1 \cdots \int_0^{\tau_{<\Gamma+1}} \mathrm{d}\tau_{\Gamma+1} \; \tau_1^{p_1} \cdots \tau_{\Gamma+1}^{p_{\Gamma+1}} = \int_0^{\tau} \mathrm{d}\tau_1 \cdots \int_0^{\tau_{<\Gamma}} \mathrm{d}\tau_\Gamma \; \frac{\tau_1^{q_1} \cdots \tau_\Gamma^{q_\Gamma}}{p_{\Gamma+1} + 1}, \tag{6.31}$$

where $q_1 + \cdots + q_\Gamma = p_1 + \cdots + p_{\Gamma+1} + 1$. The claim then follows from the inductive hypothesis. □

For most of our analysis, however, a direct calculation of order conditions is inefficient. In particular, a $(2k)$th-order Suzuki formula contains $2 \cdot 5^{k-1}$ matrix

exponentials and a direct analysis becomes prohibitive when k is large. Instead, we follow standard rules of order conditions to compute them indirectly, some of which are summarized below:

Proposition 36 (Rules of order conditions)**.** *Let $\mathscr{F}(\tau)$ and $\mathscr{G}(\tau)$ be operator-valued functions defined on $\mathbb{R}$ that are infinitely differentiable. Let p and q be nonnegative integers. The following rules of order conditions hold:*

1. *Addition: if $\mathscr{F}(\tau) = O(\tau^p)$ and $\mathscr{G}(\tau) = O(\tau^q)$, then $\mathscr{F}(\tau)+\mathscr{G}(\tau) = O(\tau^{\min(p,q)})$;*
2. *Multiplication: if $\mathscr{F}(\tau) = O(\tau^p)$ and $\mathscr{G}(\tau) = O(\tau^q)$, then $\mathscr{F}(\tau)\mathscr{G}(\tau) = O(\tau^{p+q})$;*
3. *Differentiation: $\mathscr{F}(\tau) = O(\tau^{p+1})$ if and only if $\mathscr{F}(0) = 0$ and $\mathscr{F}'(\tau) = O(\tau^p)$;*
4. *Integration: $\mathscr{F}(\tau) = O(\tau^p)$ if and only if $\int_0^t \mathrm{d}\tau \mathscr{F}(\tau) = O(t^{p+1})$; and*
5. *Exponentiation: $\mathscr{F}(\tau) = \mathscr{G}(\tau) + O(\tau^p)$ if and only if $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \mathscr{F}(\tau)\right) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \mathscr{G}(\tau)\right) + O(t^{p+1})$.*

Proof. We only prove the exponentiation rule, as the other rules follow directly from Lemma 34. Suppose that $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\, \mathscr{F}(\tau)\right) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\, \mathscr{G}(\tau)\right) + O(t^{p+1})$. To prove $\mathscr{F}(\tau) = \mathscr{G}(\tau) + O(\tau^p)$, it suffices to show that $\mathscr{F}^{(q)}(0) = \mathscr{G}^{(q)}(0)$ for $q = 0, \ldots, p-1$.

We prove this by induction. By the differentiation rule, we have

$$\mathscr{F}(t)\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{F}(\tau)\right) = \mathscr{G}(t)\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{G}(\tau)\right) + O(t^p), \tag{6.32}$$

so Lemma 34 implies $\mathscr{F}(0) = \mathscr{G}(0)$. This proves the claim in the base case. Now assume that $\mathscr{F}^{(l)}(0) = \mathscr{G}^{(l)}(0)$ holds for $l = 0, \dots q$, where $q < p-1$. By Lemma 34 and the general Leibniz rule,

$$\begin{aligned}&\sum_{l=0}^{q+1}\binom{q+1}{l}\mathscr{F}^{q+1-l}(0)\exp_{\mathcal{T}}^{(l)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{F}(\tau)\right)\\ &=\sum_{l=0}^{q+1}\binom{q+1}{l}\mathscr{G}^{q+1-l}(0)\exp_{\mathcal{T}}^{(l)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{G}(\tau)\right).\end{aligned} \tag{6.33}$$

Lemma 34 also implies $\exp_{\mathcal{T}}^{(l)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{F}(\tau)\right) = \exp_{\mathcal{T}}^{(l)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{G}(\tau)\right)$ for $l = 0, \dots, q+1$. So the above equation simplifies to

$$\mathscr{F}^{(q+1)}(0) = \mathscr{G}^{(q+1)}(0). \tag{6.34}$$

This completes the inductive step.

For the reverse direction, we want to prove $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{F}(\tau)\right) = \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\ \mathscr{G}(\tau)\right) + O(t^{p+1})$ assuming that $\mathscr{F}(\tau) = \mathscr{G}(\tau) + O(\tau^p)$. Equivalently, we want to show that $\exp_{\mathcal{T}}^{(q+1)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{F}(\tau)\right) = \exp_{\mathcal{T}}^{(q+1)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{G}(\tau)\right)$ for $q = 0, \dots, p-1$ given that $\mathscr{F}^{(q)}(0) = \mathscr{G}^{(q)}(0)$. This can be proved by induction and by applying

the Leibniz rule in a similar way as above. Specifically, the base case follows from

$$\exp_{\mathcal{T}}^{(1)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{F}(\tau)\right) = \mathscr{F}(0) = \mathscr{G}(0) = \exp_{\mathcal{T}}^{(1)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{G}(\tau)\right) \tag{6.35}$$

and the inductive step follows from

$$\begin{aligned}
\exp_{\mathcal{T}}^{(q+1)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{F}(\tau)\right) &= \sum_{l=0}^{q} \binom{q}{l} \mathscr{F}^{(q-l)}(0) \exp_{\mathcal{T}}^{(l)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{F}(\tau)\right) \\
&= \sum_{l=0}^{q} \binom{q}{l} \mathscr{G}^{(q-l)}(0) \exp_{\mathcal{T}}^{(l)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{G}(\tau)\right) \\
&= \exp_{\mathcal{T}}^{(q+1)}\left(\int_0^0 \mathrm{d}\tau\ \mathscr{G}(\tau)\right).
\end{aligned} \tag{6.36}$$

□

We now compute order conditions for the additive, multiplicative, and exponentiated Trotter error. In Section 6.5, we apply these conditions to cancel low-order Trotter error terms and represent higher-order ones as nested commutators of operator summands.

Theorem 37 (Order conditions of Trotter error)**.** *Let H be an operator, and let $\mathscr{S}(\tau)$, $\mathscr{T}(\tau)$, $\mathscr{E}(\tau)$, and $\mathscr{M}(\tau)$ be infinitely differentiable operator-valued functions*

defined for $\tau \in \mathbb{R}$*, such that*

$$\begin{aligned}\mathscr{S}(t) &= e^{tH} + \int_0^t \mathrm{d}\tau\ e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau),\\ &= \exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau\big(H + \mathscr{E}(\tau)\big)\right),\\ &= e^{tH}(I + \mathscr{M}(t)).\end{aligned} \tag{6.37}$$

For any nonnegative integer p*, the following conditions are equivalent:*

1. $\mathscr{S}(t) = e^{tH} + O\left(t^{p+1}\right)$*;*
2. $\mathscr{T}(\tau) = O\left(\tau^{p}\right)$*;*
3. $\mathscr{E}(\tau) = O\left(\tau^{p}\right)$*; and*
4. $\mathscr{M}(t) = O\left(t^{p+1}\right)$*.*

Proof. Suppose that $\mathscr{T}(\tau) = O(\tau^p)$. We apply the multiplication rule of Proposition 36 to get $e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau) = O(\tau^p)$. A further application of the integration rule gives $\mathscr{S}(t) - e^{tH} = \int_0^t \mathrm{d}\tau\ e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau) = O(t^{p+1})$.

Conversely, let $\mathscr{S}(t) = e^{tH} + O(t^{p+1})$. This implies $\int_0^t \mathrm{d}\tau\ e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau) = O(t^{p+1})$. Applying the integration rule and the multiplication rule gives $\mathscr{S}(\tau)\mathscr{T}(\tau) = O(\tau^p)$. Note that $\mathscr{S}(t) = e^{tH} + O(t^{p+1}) = I + O(t)$ implies that the operator-valued function $\mathscr{S}(t)$ is invertible for sufficiently small t and, since $\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}^{-1}(t) = -\mathscr{S}^{-1}(t)\mathscr{S}'(t)\mathscr{S}^{-1}(t)$, the inverse function $\mathscr{S}^{-1}(t)$ is infinitely differentiable. Applying the multiplication rule gives $\mathscr{T}(\tau) = O(\tau^p)$, which establishes the equivalence of Conditions 1 and 2.

Note that $\mathscr{S}(t) = e^{tH} + O(t^{p+1})$ is equivalent to $\exp_{\mathcal{T}}\big(\int_0^t \mathrm{d}\tau(H + \mathscr{E}(\tau))\big) = e^{tH} + O(t^{p+1})$, which is further equivalent to $H + \mathscr{E}(\tau) = H + O(\tau^p)$ by the exponentiation rule. Canceling H from both sides proves the equivalence of Conditions 1 and 3.

Finally, note that $\mathscr{S}(t) = e^{tH}(I + \mathscr{M}(t)) = e^{tH} + O(t^{p+1})$ can be simplified to $e^{tH}\mathscr{M}(t) = O(t^{p+1})$. The equivalence of Conditions 1 and 4 then follows from the multiplication rule. □

6.5 Error representations

For a product formula with a certain error type and order condition, we now represent its error in terms of nested commutators of the operator summands.

Consider an operator $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ with Γ summands. The ideal evolution generated by H is e^{tH}, which we decompose using a pth-order product formula $\mathscr{S}(t) = \prod_{\upsilon=1}^{\Upsilon}\prod_{\gamma=1}^{\Gamma} e^{ta_{(\upsilon,\gamma)}H_{\pi_\upsilon(\gamma)}}$. We know from Theorem 33 that the Trotter error can be expressed in the additive form $\mathscr{S}(t) = e^{tH} + \int_0^t \mathrm{d}\tau\, e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau)$, the multiplicative form $\mathscr{S}(t) = e^{tH}(I + \mathscr{M}(t))$, where $\mathscr{M}(t) = \exp_{\mathcal{T}}\big(\int_0^t \mathrm{d}\tau\, e^{-\tau H}\mathscr{E}(\tau)e^{\tau H}\big) - I$, and the exponentiated form $\mathscr{S}(t) = \exp_{\mathcal{T}}\big(\int_0^t \mathrm{d}\tau\big(H + \mathscr{E}(\tau)\big)\big)$. Furthermore, both $\mathscr{T}(\tau)$ and $\mathscr{E}(\tau)$ consist of conjugations of matrix exponentials and have order condition $\mathscr{T}(\tau),\ \mathscr{E}(\tau) \in O(\tau^p)$ (Theorem 37).

We first consider the representation of a single conjugation of matrix expo-

nentials

$$e^{\tau A_s} \cdots e^{\tau A_2} e^{\tau A_1} B e^{-\tau A_1} e^{-\tau A_2} \cdots e^{-\tau A_s}, \tag{6.38}$$

where $A_1, A_2, \ldots, A_s, B$ are operators and $\tau \in \mathbb{R}$. Our goal is to expand this conjugation into a finite series in the time variable τ. We will only keep track of those terms with order $O(\tau^p)$, because terms corresponding to $1, \tau, \ldots, \tau^{p-1}$ will vanish in the final representation of Trotter error due to the order condition. As mentioned before, such a conjugation was previously analyzed based on a naive application of the Taylor's theorem [36] and an infinite-series expansion [98]. However, those results do not represent Trotter error as a finite number of commutators of operator summands and they only apply to special systems such as those with geometrical locality or suitable Lie-algebraic structure. Our new representation overcomes these limitations.

We begin with the innermost layer $e^{\tau A_1} B e^{-\tau A_1}$. Applying Taylor's theorem to order $p-1$ with integral form of the remainder, we have

$$\begin{aligned} e^{\tau A_1} B e^{-\tau A_1} = B + \big[A_1, B\big]\tau + \cdots + \underbrace{\big[A_1, \cdots, \big[A_1}_{p-1}, B\big] \cdots \big] \frac{\tau^{p-1}}{(p-1)!} \\ + \int_0^\tau \mathrm{d}\tau_2 \; e^{(\tau-\tau_2)A_1} \underbrace{\big[A_1, \cdots, \big[A_1}_{p}, B\big] \cdots \big] e^{-(\tau-\tau_2)A_1} \frac{\tau_2^{p-1}}{(p-1)!}. \end{aligned} \tag{6.39}$$

Using the abbreviation $\mathrm{ad}_{A_1}(B) = \big[A_1, B\big]$, we rewrite

$$\begin{aligned} e^{\tau A_1} B e^{-\tau A_1} &= B + \mathrm{ad}_{A_1}(B)\tau + \cdots + \mathrm{ad}_{A_1}^{p-1}(B)\frac{\tau^{p-1}}{(p-1)!} \\ &\quad + \int_0^\tau \mathrm{d}\tau_2 \; e^{(\tau-\tau_2)A_1}\mathrm{ad}_{A_1}^{p}(B)e^{-(\tau-\tau_2)A_1}\frac{\tau_2^{p-1}}{(p-1)!}. \end{aligned} \tag{6.40}$$

By the multiplication rule and the integration rule of Proposition 36, the last term has order

$$\int_0^\tau \mathrm{d}\tau_2 \; e^{(\tau-\tau_2)A_1}\mathrm{ad}_{A_1}^{p}(B)e^{-(\tau-\tau_2)A_1}\frac{\tau_2^{p-1}}{(p-1)!} = O\left(\tau^p\right). \tag{6.41}$$

This term cannot be canceled by the order condition, and we keep it in our expansion. The remaining terms corresponding to $1, \tau, \ldots, \tau^{p-1}$ are substituted back to the original conjugation of matrix exponentials.

We now consider the next layer of conjugation. We apply Taylor's theorem to the operators $e^{\tau A_2} B e^{-\tau A_2}$, $e^{\tau A_2}\mathrm{ad}_{A_1}(B)e^{-\tau A_2}$, $\ldots$, $e^{\tau A_2}\mathrm{ad}_{A_1}^{p-1}(B)e^{-\tau A_2}$ to order

$p-1$, $p-2$, ..., 0, respectively, obtaining

$$\begin{aligned}
e^{\tau A_2} B e^{-\tau A_2} &= B + \cdots + \mathrm{ad}_{A_2}^{p-1}(B)\frac{\tau^{p-1}}{(p-1)!} + \int_0^\tau \mathrm{d}\tau_2 \\
&\qquad e^{(\tau-\tau_2)A_2}\mathrm{ad}_{A_2}^{p}(B)e^{-(\tau-\tau_2)A_2}\frac{\tau_2^{p-1}}{(p-1)!}, \\
e^{\tau A_2}\mathrm{ad}_{A_1}(B)e^{-\tau A_2} &= \mathrm{ad}_{A_1}(B) + \cdots + \mathrm{ad}_{A_2}^{p-2}\mathrm{ad}_{A_1}(B)\frac{\tau^{p-2}}{(p-2)!} \\
&\quad + \int_0^\tau \mathrm{d}\tau_2\; e^{(\tau-\tau_2)A_2}\mathrm{ad}_{A_2}^{p-1}\mathrm{ad}_{A_1}(B)e^{-(\tau-\tau_2)A_2}\frac{\tau_2^{p-2}}{(p-2)!}, \\
&\vdots \\
e^{\tau A_2}\mathrm{ad}_{A_1}^{p-1}(B)e^{-\tau A_2} &= \mathrm{ad}_{A_1}^{p-1}(B) + \int_0^\tau \mathrm{d}\tau_2\; e^{(\tau-\tau_2)A_2}\mathrm{ad}_{A_2}\mathrm{ad}_{A_1}^{p-1}(B)e^{-(\tau-\tau_2)A_2}.
\end{aligned} \tag{6.42}$$

Combining with the result from the first layer, the Taylor remainders in the above equation have order

$$\begin{aligned}
\int_0^\tau \mathrm{d}\tau_2\; e^{(\tau-\tau_2)A_2}\mathrm{ad}_{A_2}^{p}(B)e^{-(\tau-\tau_2)A_2}\frac{\tau_2^{p-1}}{(p-1)!} &= O\left(\tau^p\right), \\
\int_0^\tau \mathrm{d}\tau_2\; e^{(\tau-\tau_2)A_2}\mathrm{ad}_{A_2}^{p-1}\mathrm{ad}_{A_1}(B)e^{-(\tau-\tau_2)A_2}\frac{\tau_2^{p-2}}{(p-2)!}\tau &= O\left(\tau^p\right), \\
&\vdots \\
\int_0^\tau \mathrm{d}\tau_2\; e^{(\tau-\tau_2)A_2}\mathrm{ad}_{A_2}\mathrm{ad}_{A_1}^{p-1}(B)e^{-(\tau-\tau_2)A_2}\frac{\tau^{p-1}}{(p-1)!} &= O\left(\tau^p\right).
\end{aligned} \tag{6.43}$$

We keep these terms in our expansion and substitute the remaining ones back to the original conjugation of matrix exponentials.

We repeat this analysis for all the remaining layers of the conjugation of matrix exponentials. In doing so, we keep track of those terms with order $O\left(\tau^p\right)$,

obtaining

$$
\begin{aligned}
&e^{\tau A_s}\cdots e^{\tau A_2}e^{\tau A_1}Be^{-\tau A_1}e^{-\tau A_2}\cdots e^{-\tau A_s}\\
&= C_0 + C_1\tau + \cdots + C_{p-1}\tau^{p-1}\\
&\quad + \sum_{k=1}^{s}\sum_{\substack{q_1+\cdots+q_k=p\\ q_k\neq 0}} e^{\tau A_s}\cdots e^{\tau A_{k+1}}\\
&\quad \cdot \int_0^\tau \mathrm{d}\tau_2\; e^{\tau_2 A_k}\mathrm{ad}_{A_k}^{q_k}\cdots\mathrm{ad}_{A_1}^{q_1}(B)e^{-\tau_2 A_k}\cdot\frac{(\tau-\tau_2)^{q_k-1}\tau^{q_1+\cdots+q_{k-1}}}{(q_k-1)!q_{k-1}!\cdots q_1!}\\
&\quad \cdot e^{-\tau A_{k+1}}\cdots e^{-\tau A_s}
\end{aligned}
\tag{6.44}
$$

for some operators $C_0, C_1, \ldots, C_{p-1}$. Due to the order condition, the terms of order $1, \tau, \ldots, \tau^{p-1}$ will vanish in our final representation of the Trotter error.

We now bound the spectral norm of those terms with order $O(\tau^p)$. By the triangle inequality, we have an upper bound of

$$
\begin{aligned}
&\sum_{k=1}^{s}\sum_{\substack{q_1+\cdots+q_k=p\\ q_k\neq 0}}\int_0^{|\tau|}\mathrm{d}\tau_2\;\frac{(|\tau|-\tau_2)^{q_k-1}\,|\tau|^{q_1+\cdots+q_{k-1}}}{(q_k-1)!q_{k-1}!\cdots q_1!}\left\|\mathrm{ad}_{A_k}^{q_k}\cdots\mathrm{ad}_{A_1}^{q_1}(B)\right\|e^{2|\tau|\sum_{l=1}^{s}\|A_l\|}\\
&= \sum_{k=1}^{s}\sum_{\substack{q_1+\cdots+q_k=p\\ q_k\neq 0}}\binom{p}{q_1\;\cdots\;q_k}\frac{|\tau|^p}{p!}\left\|\mathrm{ad}_{A_k}^{q_k}\cdots\mathrm{ad}_{A_1}^{q_1}(B)\right\|e^{2|\tau|\sum_{l=1}^{s}\|A_l\|}\\
&= \sum_{q_1+\cdots+q_s=p}\binom{p}{q_1\;\cdots\;q_s}\frac{|\tau|^p}{p!}\left\|\mathrm{ad}_{A_s}^{q_s}\cdots\mathrm{ad}_{A_1}^{q_1}(B)\right\|e^{2|\tau|\sum_{l=1}^{s}\|A_l\|}\\
&= \alpha_{\mathrm{comm}}\big(A_s,\ldots,A_1,B\big)\frac{|\tau|^p}{p!}e^{2|\tau|\sum_{l=1}^{s}\|A_l\|},
\end{aligned}
\tag{6.45}
$$

where

$$\alpha_{\mathrm{comm}}(A_s, \ldots, A_1, B) := \sum_{q_1+\cdots+q_s=p} \binom{p}{q_1 \ \cdots \ q_s} \left\| \mathrm{ad}_{A_s}^{q_s} \cdots \mathrm{ad}_{A_1}^{q_1}(B) \right\|. \tag{6.46}$$

This bound holds for arbitrary operators $A_1, A_2, \ldots, A_s$. When these operators are anti-Hermitian, we can tighten the above analysis by evaluating the spectral norm of a matrix exponential as 1. We have therefore established:

Theorem 38 (Commutator expansion of a conjugation of matrix exponentials)**.** *Let $A_1, A_2, \ldots, A_s$ and B be operators. Then the conjugation $e^{\tau A_s} \cdots e^{\tau A_2} e^{\tau A_1} B e^{-\tau A_1} e^{-\tau A_2} \cdots e^{-\tau A_s}$ $(\tau \in \mathbb{R})$ has the expansion*

$$e^{\tau A_s} \cdots e^{\tau A_2} e^{\tau A_1} B e^{-\tau A_1} e^{-\tau A_2} \cdots e^{-\tau A_s} = C_0 + C_1 \tau + \cdots + C_{p-1} \tau^{p-1} + \mathscr{C}(\tau). \tag{6.47}$$

Here, $C_0, \ldots, C_{p-1}$ are operators independent of τ. The operator-valued function $\mathscr{C}(\tau)$ is given by

$$\begin{aligned} \mathscr{C}(\tau) := \sum_{k=1}^{s} \sum_{\substack{q_1+\cdots+q_k=p \\ q_k \neq 0}} & e^{\tau A_s} \cdots e^{\tau A_{k+1}} \\ & \cdot \int_0^\tau \mathrm{d}\tau_2 \ e^{\tau_2 A_k} \mathrm{ad}_{A_k}^{q_k} \cdots \mathrm{ad}_{A_1}^{q_1}(B) e^{-\tau_2 A_k} \cdot \frac{(\tau - \tau_2)^{q_k - 1} \tau^{q_1 + \cdots + q_{k-1}}}{(q_k - 1)! q_{k-1}! \cdots q_1!} \\ & \cdot e^{-\tau A_{k+1}} \cdots e^{-\tau A_s}. \end{aligned} \tag{6.48}$$

Furthermore, we have the spectral-norm bound

$$\|\mathscr{C}(\tau)\| \leq \alpha_{\mathrm{comm}}\big(A_s, \ldots, A_1, B\big) \frac{|\tau|^p}{p!} e^{2|\tau| \sum_{k=1}^s \|A_k\|} \tag{6.49}$$

for general operators and

$$\|\mathscr{C}(\tau)\| \leq \alpha_{\mathrm{comm}}\big(A_s, \ldots, A_1, B\big) \frac{|\tau|^p}{p!} \tag{6.50}$$

when A_k $(k = 1, \ldots, s)$ are anti-Hermitian, where

$$\alpha_{\mathrm{comm}}\big(A_s, \ldots, A_1, B\big) = \sum_{q_1 + \cdots + q_s = p} \binom{p}{q_1 \ \cdots \ q_s} \left\| \mathrm{ad}_{A_s}^{q_s} \cdots \mathrm{ad}_{A_1}^{q_1}(B) \right\|. \tag{6.51}$$

We now apply the above theorem to analyze Trotter error. For simplicity, we only consider the additive error, although the analysis can be easily adapted to handle the multiplicative error and the exponentiated error.

Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be an operator that generates the evolution $e^{tH} = e^{t \sum_{\gamma=1}^{\Gamma} H_\gamma}$. Let $\mathscr{S}(t) = \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}$ be a pth-order product formula as in Section 2.4. We know from Theorem 33 that the Trotter error can be expressed in an

additive form as $\mathscr{S}(t) = e^{tH} + \int_0^t \mathrm{d}\tau\ e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau)$, where

$$\begin{aligned}\mathscr{T}(\tau) = &\sum_{(\upsilon,\gamma)} \overrightarrow{\prod_{(\upsilon',\gamma')\prec(\upsilon,\gamma)}} e^{-\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \left(a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}\right) \overleftarrow{\prod_{(\upsilon',\gamma')\prec(\upsilon,\gamma)}} e^{\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} \\ &- \overrightarrow{\prod_{(\upsilon',\gamma')}} e^{-\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}} H \overleftarrow{\prod_{(\upsilon',\gamma')}} e^{\tau a_{(\upsilon',\gamma')} H_{\pi_{\upsilon'}(\gamma')}}.\end{aligned} \tag{6.52}$$

Furthermore, Theorem 37 implies that the operator-valued function $\mathscr{T}(\tau)$ satisfies the order condition $\mathscr{T}(\tau) = O(\tau^p)$.

We now apply Theorem 38 to expand every conjugation of matrix exponentials in $\mathscr{T}(\tau)$. In doing so, we only keep track of terms of order $O(\tau^p)$, as those terms corresponding to $1, \tau, \ldots, \tau^{p-1}$ will vanish due to the order condition. We obtain

$$\begin{aligned}&\|\mathscr{T}(\tau)\| \\ &\leq \sum_{(\upsilon,\gamma)} \alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}, (\upsilon',\gamma') \prec (\upsilon,\gamma)\right\}}, H_{\pi_\upsilon(\gamma)}\right) \frac{\tau^p}{p!} \exp\left(2\tau \sum_{(\upsilon',\gamma')\prec(\upsilon,\gamma)} \left\|H_{\pi_{\upsilon'}(\gamma')}\right\|\right) \\ &\quad + \alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}\right\}}, H\right) \frac{\tau^p}{p!} \exp\left(2\tau \sum_{(\upsilon',\gamma')} \left\|H_{\pi_{\upsilon'}(\gamma')}\right\|\right),\end{aligned} \tag{6.53}$$

where $\overrightarrow{\{\}}$ denotes an ordered list where elements have increasing indices from left to right. This is further bounded by

$$\begin{aligned}\|\mathscr{T}(\tau)\| &\leq 2\sum_{(\upsilon,\gamma)} \alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}\right\}}, H_{\pi_\upsilon(\gamma)}\right) \frac{\tau^p}{p!} \exp\left(2\tau \sum_{(\upsilon',\gamma')} \left\|H_{\pi_{\upsilon'}(\gamma')}\right\|\right) \\ &= 2\Upsilon \sum_{\gamma=1}^{\Gamma} \alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}\right\}}, H_\gamma\right) \frac{\tau^p}{p!} \exp\left(2\tau\Upsilon \sum_{\gamma'=1}^{\Gamma} \|H_{\gamma'}\|\right).\end{aligned} \tag{6.54}$$

After a final integration over τ, we have

$$\begin{aligned}\left\|\mathscr{S}(t)-e^{tH}\right\| &\leq \int_0^t \mathrm{d}\tau \left\|e^{(t-\tau)H}\mathscr{S}(\tau)\mathscr{T}(\tau)\right\| \\ &\leq 2\Upsilon \sum_{\gamma=1}^{\Gamma} \alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}\right\}}, H_\gamma\right) \frac{t^{p+1}}{(p+1)!} \exp\left(4t\Upsilon \sum_{\gamma'=1}^{\Gamma} \|H_{\gamma'}\|\right).\end{aligned} \tag{6.55}$$

The factor 4 in the above bound can be tightened to 2 by directly substituting (6.48) into (6.52), giving

$$\left\|\mathscr{S}(t)-e^{tH}\right\| \leq 2\Upsilon \sum_{\gamma=1}^{\Gamma} \alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}\right\}}, H_\gamma\right) \frac{t^{p+1}}{(p+1)!} \exp\left(2t\Upsilon \sum_{\gamma'=1}^{\Gamma} \|H_{\gamma'}\|\right). \tag{6.56}$$

This bound holds for arbitrary operators H_γ. If the operator summands are anti-Hermitian, the bound can be further tightened to

$$\left\|\mathscr{S}(t)-e^{tH}\right\| \leq 2\Upsilon \sum_{\gamma=1}^{\Gamma} \alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}\right\}}, H_\gamma\right) \frac{t^{p+1}}{(p+1)!}. \tag{6.57}$$

Note that our analysis depends on $\pi_{\upsilon'}$, the ordering of operator summands in stage υ' of the product formula. In the following, we prove an asymptotic bound that removes this ordering constraint. The resulting bound is independent of the definition of product formula and may thus be easier to compute in practice. Our analysis here is not tight in terms of the constant prefactor, but it is sufficient to establish the desired commutator scaling.

Recall from Theorem 38 that

$$\alpha_{\mathrm{comm}}\left(\overrightarrow{\left\{H_{\pi_{\upsilon'}(\gamma')}\right\}}, H_\gamma\right) = \sum_{q_{(1,1)}+\cdots+q_{(\Upsilon,\Gamma)}=p} \binom{p}{q_{(1,1)} \ \cdots \ q_{(\Upsilon,\Gamma)}} \left\|\mathrm{ad}_{H_{\pi_1(1)}}^{q_{(1,1)}} \cdots \mathrm{ad}_{H_{\pi_\Upsilon(\Gamma)}}^{q_{(\Upsilon,\Gamma)}}(H_\gamma)\right\|, \tag{6.58}$$

which is at most $p!$ times $\sum_{q_{(1,1)}+\cdots+q_{(\Upsilon,\Gamma)}=p} \left\|\mathrm{ad}_{H_{\pi_1(1)}}^{q_{(1,1)}} \cdots \mathrm{ad}_{H_{\pi_\Upsilon(\Gamma)}}^{q_{(\Upsilon,\Gamma)}}(H_\gamma)\right\|$. Fixing the value of γ, we claim that

$$\sum_{q_{(1,1)}+\cdots+q_{(\Upsilon,\Gamma)}=p} \left\|\mathrm{ad}_{H_{\pi_1(1)}}^{q_{(1,1)}} \cdots \mathrm{ad}_{H_{\pi_\Upsilon(\Gamma)}}^{q_{(\Upsilon,\Gamma)}}(H_\gamma)\right\| \leq \Upsilon^p \sum_{\gamma_{p+1}=1}^{\Gamma} \cdots \sum_{\gamma_2=1}^{\Gamma} \left\|\left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_\gamma\right]\right]\right\|. \tag{6.59}$$

This can be seen as follows. Every nested commutator on the left-hand side has p nesting layers and must thus be of the form on the right. Conversely, we fix one term $\left\|\left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_\gamma\right]\right]\right\|$ from the right and bound the number of times this term might appear on the left. Each operator $H_{\gamma_2}, \ldots, H_{\gamma_{p+1}}$ can appear in Υ possible stages and hence there are Υ^p possibilities in total. When the stages are fixed, this will uniquely determine one term $\left\|\mathrm{ad}_{H_{\pi_1(1)}}^{q_{(1,1)}} \cdots \mathrm{ad}_{H_{\pi_\Upsilon(\Gamma)}}^{q_{(\Upsilon,\Gamma)}}(H_\gamma)\right\|$ on the left. We have thus established the commutator scaling of Trotter error.

Theorem 39 (Trotter error with commutator scaling)**.** *Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be an operator consisting of Γ summands and $t \geq 0$. Let $\mathscr{S}(t) = \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}$ be a pth-order formula. Define $\widetilde{\alpha}_{\mathrm{comm}} = \sum_{\gamma_1,\gamma_2,\ldots,\gamma_{p+1}=1}^{\Gamma} \left\|\left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right]\right\|$. Then, the additive Trotter error and the multiplicative Trotter error, defined respectively by $\mathscr{S}(t) = e^{tH} + \mathscr{A}(t)$ and $\mathscr{S}(t) = e^{tH}(I + \mathscr{M}(t))$, can be asymptotically*

bounded as

$$\|\mathscr{A}(t)\| = \mathcal{O}\left(\widetilde{\alpha}_{\text{comm}} t^{p+1} e^{2t\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|}\right), \quad \|\mathscr{M}(t)\| = \mathcal{O}\left(\widetilde{\alpha}_{\text{comm}} t^{p+1} e^{2t\Upsilon \sum_{\gamma=1}^{\Gamma} \|H_\gamma\|}\right). \tag{6.60}$$

Furthermore, if H_γ are anti-Hermitian,

$$\|\mathscr{A}(t)\| = \mathcal{O}\left(\widetilde{\alpha}_{\text{comm}} t^{p+1}\right), \quad \|\mathscr{M}(t)\| = \mathcal{O}\left(\widetilde{\alpha}_{\text{comm}} t^{p+1}\right). \tag{6.61}$$

Corollary 40 (Trotter number with commutator scaling)**.** *Let $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ be an operator consisting of Γ summands with H_γ anti-Hermitian and $t \geq 0$. Let $\mathscr{S}(t) = \prod_{\upsilon=1}^{\Upsilon} \prod_{\gamma=1}^{\Gamma} e^{t a_{(\upsilon,\gamma)} H_{\pi_\upsilon(\gamma)}}$ be a pth-order product formula. Define $\widetilde{\alpha}_{\text{comm}} = \sum_{\gamma_1,\gamma_2,\ldots,\gamma_{p+1}=1}^{\Gamma} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right] \right\|$. Then, we have $\left\|\mathscr{S}^r(t/r) - e^{tH}\right\| = \mathcal{O}(\epsilon)$ provided*

$$r = \mathcal{O}\left(\frac{\widetilde{\alpha}_{\text{comm}}^{1/p} t^{1+1/p}}{\epsilon^{1/p}}\right). \tag{6.62}$$

For any $\delta > 0$, we can choose p sufficiently large so that $1/p < \delta$. For this choice of p, we have $r = \mathcal{O}\left(\widetilde{\alpha}_{\text{comm}}^{\delta} t^{1+\delta}/\epsilon^{\delta}\right)$. Therefore, the Trotter number scales as $r = \widetilde{\alpha}_{\text{comm}}^{o(1)} t^{1+o(1)}$ if we simulate with constant accuracy. To obtain the asymptotic complexity of the product-formula algorithm, it thus suffices to compute the quantity $\widetilde{\alpha}_{\text{comm}} = \sum_{\gamma_1,\gamma_2,\ldots,\gamma_{p+1}} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right] \right\|$, which can often be done by induction. We illustrate this by presenting a host of applications of our bound to simulating quantum dynamics (Chapter 7) and quantum Monte Carlo methods

(Chapter 9).

Note that we did not evaluate the constant prefactor of our bound in Theorem 39. For that purpose, it is better to use Theorem 38, which gives a concrete expression for the error operator. We provide numerical evidence in Chapter 7 suggesting that our bound has a small prefactor.

Chapter 7: Analysis of product formulas: concrete systems

Our result on the analysis of Trotter error, in particular the commutator scaling of Trotter error (Theorem 39), uncovers a host of new speedups of the product-formula algorithm. In this chapter, we analyze the performance of product formulas for simulating concrete physical systems, including nearest-neighbor lattice systems (Section 7.1), electronic structure Hamiltonians (Section 7.2), k-local Hamiltonians (Section 7.3), rapidly decaying power-law interactions (Section 7.4), and clustered Hamiltonians (Section 7.5). Our result nearly matches or even outperforms the best previous results in digital quantum simulation. We accompany our analysis with numerical calculation in Section 7.6, which suggests that the error bounds also have nearly tight constant prefactors. We conclude in Section 7.7 with a brief discussion of the results and some open questions.

This chapter is partly based on the following papers:

[36] Andrew M. Childs and Yuan Su, *Nearly optimal lattice simulation by product formulas*, Physical Review Letters **123** (2019), 050503, arXiv:1901.00564.

[37] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu, *A theory of Trotter error*, 2019, arXiv:1912.08854.

7.1 Nearest-neighbor lattice Hamiltonians

A natural class of Hamiltonians that includes many physically reasonable systems is the class of lattice Hamiltonians [49, 53, 64, 78]. Lattice Hamiltonians arise in many models of condensed matter physics, including systems of spins (e.g., Ising, XY, and Heisenberg models; Kitaev's toric code and honeycomb models; etc.), fermions (e.g., the Hubbard model and the t-J model), and bosons (e.g., the Bose-Hubbard model). Note that fermion models can be simulated using local interactions among qubits by using a mapping to qubits that preserves locality [103]. Digital simulations of quantum field theory also typically involve approximation by a lattice system [64].

For simplicity, we mainly focus on nearest-neighbor lattice systems in one dimension (although the analysis can be generalized to other lattice models as well). In this case, n qubits are laid out on a one-dimensional lattice and the Hamiltonian only involves nearest-neighbor interactions. Specifically, a Hamiltonian H is a lattice Hamiltonian if it acts on n qubits and can be decomposed as $H = \sum_{j=1}^{n-1} H_{j,j+1}$, where each $H_{j,j+1}$ is a Hermitian operator that acts nontrivially only on qubits j and $j+1$. We assume that $\max_j \|H_{j,j+1}\| \leq 1$, for otherwise we evolve under the normalized Hamiltonian $H/\max_j \|H_{j,j+1}\|$ for time $\max_j \|H_{j,j+1}\| t$.

As established in Theorem 39, the asymptotic performance of a pth-order

product formula depends on the quantity

$$\widetilde{\alpha}_{\text{comm}} = \sum_{j_1,j_2,\ldots,j_{p+1}=1}^{n-1} \left\| \left[H_{j_{p+1},j_{p+1}+1}, \cdots \left[H_{j_2,j_2+1}, H_{j_1,j_1+1} \right] \right] \right\| . \tag{7.1}$$

For this nested commutator to be nonzero, the set of qubits on which the outer operators act must intersect with those of the inner operators. In other words, if we fix the choice of j_1, then it must hold that $j_2 \in \{j_1 - 1, j_1, j_1 + 1\}$, $j_3 \in \{j_1 - 2, \ldots, j_1 + 2\}$,..., $j_{p+1} \in \{j_1 - p, \ldots, j_1 + p\}$. Therefore, we estimate

$$\widetilde{\alpha}_{\text{comm}} \leq 2^p \cdot 3 \cdot 5 \cdots (2p+1) \left(\max_j \|H_{j,j+1}\| \right)^p \sum_{j_1=1}^{n-1} \|H_{j_1,j_1+1}\| = \mathcal{O}(n) . \tag{7.2}$$

Corollary 40 then implies that a Trotter number of $r = \mathcal{O}\left(n^{1/p} t^{1+1/p} / \epsilon^{1/p}\right)$ suffices to simulate for time t with accuracy ϵ. Choosing p sufficiently large, letting ϵ be constant, and implementing each Trotter step with $\mathcal{O}(n)$ gates, we have the gate complexity

$$(nt)^{1+o(1)} \tag{7.3}$$

for simulating nearest-neighbor lattice Hamiltonians.

Based on the intuition from the BCH expansion, Jordan, Lee, and Preskill claimed that product formulas can simulate an n-qubit lattice Hamiltonian for time t using only $(nt)^{1+o(1)}$ gates [64], but they did not provide rigorous justification and it is unclear how to formalize their argument. Our result gives, for the first time, a rigorous proof of the Jordan-Lee-Preskill claim, providing a nearly

optimal approach to lattice simulation simpler than the algorithm of [53] based on Lieb-Robinson bounds. As a side application, we obtain a tensor network representation of lattice systems with bond dimension $2^{n^{o(1)}t^{1+o(1)}}$, generalizing a recent construction of [59, Lemma 17].

7.2 Second-quantized electronic structure

Simulating electronic structure Hamiltonians is one of the most widely studied applications of digital quantum simulation. An efficient solution of this problem could help design and engineer new pharmaceuticals, catalysts, and materials [9]. Recent studies have focused on solving this problem using more advanced simulation algorithms. Here, we demonstrate the power of product formulas for simulating electronic structure Hamiltonians.

We consider the second-quantized representation of the electronic structure problem. In the plane-wave dual basis, the electronic structure Hamiltonian has the form [9, Eq. (8)]

$$H = \underbrace{\frac{1}{2n}\sum_{j,k,\nu} \kappa_\nu^2 \cos[\kappa_\nu \cdot r_{k-j}] A_j^\dagger A_k}_{T} \underbrace{-\frac{4\pi}{\omega}\sum_{j,\iota,\nu\neq 0} \frac{\zeta_\iota \cos[\kappa_\nu \cdot (\widetilde{r}_\iota - r_j)]}{\kappa_\nu^2} N_j}_{U} + \underbrace{\frac{2\pi}{\omega}\sum_{\substack{j\neq k \\ \nu\neq 0}} \frac{\cos[\kappa_\nu \cdot r_{j-k}]}{\kappa_\nu^2} N_j N_k}_{V}, \tag{7.4}$$

where j, k range over all n orbitals and ω is the volume of the computational cell.

Following the assumptions of [9, 76], we consider the constant density case where $n/\omega = \mathcal{O}(1)$. Here, $\kappa_\nu = 2\pi\nu/\omega^{1/3}$ are n vectors of plane-wave frequencies, where ν are three-dimensional vectors of integers with elements in $[-n^{1/3}, n^{1/3}]$; r_j are the positions of electrons; ζ_ι are nuclear charges such that $\sum_\iota |\zeta_\iota| = \mathcal{O}(n)$; and $\widetilde{r}_\iota$ are the nuclear coordinates. The operators $A_j^\dagger$ and A_k are electronic creation and annihilation operators, and $N_j = A_j^\dagger A_j$ are the number operators. The potential terms U and V are already diagonalized in the plane-wave dual basis. To further diagonalize the kinetic term T, we may switch to the plane-wave basis. This is accomplished by the fermionic fast Fourier transform FFFT [9, Eq. (10)]. We have

$$H = \mathrm{FFFT}^\dagger \underbrace{\left(\frac{1}{2}\sum_\nu \kappa_\nu^2 N_\nu\right)}_{\widetilde{T}} \mathrm{FFFT} + U + V. \tag{7.5}$$

To simulate the dynamics of such a Hamiltonian for time t, the current fastest algorithms are qubitization [7, 74] with $\widetilde{\mathcal{O}}(n^3 t)$ gate complexity and small prefactor, and the interaction-picture algorithm [76] with complexity $\widetilde{\mathcal{O}}(n^2 t)$ and large prefactor. We show that higher-order product formulas can perform the same simulation with gate complexity $n^{2+o(1)}t^{1+o(1)}$. For the special case of the second-order Suzuki formula, this confirms a recent observation of Kivlichan et al. from numerical calculation [66].

Using the plane-wave basis for the kinetic operator and the plane-wave dual basis for the potential operators, we have that all terms in $\widetilde{T}$ and $U+V$ commute with each other, respectively. Then, we can decompose $e^{-it\widetilde{T}}$ and $e^{-it(U+V)}$ into

product of elementary matrix exponentials without introducing additional error, giving the product formula

$$
\begin{aligned}
&e^{-ita_{(\Upsilon,2)}T}e^{-ita_{(\Upsilon,1)}(U+V)}\cdots e^{-ita_{(1,2)}T}e^{-ita_{(1,1)}(U+V)} \\
&= \mathrm{FFFT}^{\dagger}e^{-ita_{(\Upsilon,2)}\widetilde{T}}\mathrm{FFFT}e^{-ita_{(\Upsilon,1)}(U+V)}\cdots \mathrm{FFFT}^{\dagger}e^{-ita_{(1,2)}\widetilde{T}}\mathrm{FFFT}e^{-ita_{(1,1)}(U+V)}.
\end{aligned} \tag{7.6}
$$

For practical implementation, we need to further exponentiate spin operators using a fermionic encoding, such as the Jordan-Wigner encoding. However, these implementation details do not affect the analysis of Trotter error and will thus be ignored in the our discussion. The fermionic fast Fourier transform and the exponentiation of $\widetilde{T}$, U, and V can all be implemented using the Jordan-Wigner encoding with complexity $\widetilde{O}(n)$ [46, 76].

To analyze the performance of product formulas, we need to bound the spectral norm of the nested commutators $[H_{\gamma_{p+1}}, \cdots [H_{\gamma_2}, H_{\gamma_1}]]$, where $H_\gamma \in \{T, U, V\}$. This can be done by induction. In the base case, we need to estimate the norm of the kinetic operator T and the potential operators U and V. For readability, we use the abbreviated representation

$$
T = \sum_{j,k} t_{j,k} A_j^{\dagger} A_k, \qquad U = \sum_j u_j N_j, \qquad V = \sum_{j,k} v_{j,k} N_j N_k. \tag{7.7}
$$

Since $\left\|A_j^{\dagger}\right\| = \|A_j\| = \|N_j\| = 1$, we can apply the triangle inequality and upper bound $\|T\|$, $\|U\|$, and $\|V\|$ by the vector 1-norm $\left\|\vec{t}\right\|_1$, $\|\vec{u}\|_1$, and $\|\vec{v}\|_1$. We analyze

this in Proposition 42.

Lemma 41 ([9, (F6) and (F13)])**.** *Let an electronic structure Hamiltonian be given as in (7.4). The following asymptotic analyses hold:*

1.
$$\sum_{\nu \neq 0} \frac{1}{\kappa_\nu^2} = \mathcal{O}(n). \tag{7.8}$$

2. *For any fixed j,*
$$\sum_{\nu} \kappa_\nu^2 \cos[\kappa_\nu \cdot r_j] = \mathcal{O}(1). \tag{7.9}$$

3.
$$\sum_{\iota} |\zeta_\iota| = \mathcal{O}(n). \tag{7.10}$$

Proposition 42. *Let an electronic structure Hamiltonian be given as in (7.4). We have the following bounds on the vector 1-norm and ∞-norm of the coefficients of the kinetic operator and the potential operators:*

$$\begin{aligned} \left\|\vec{t}\right\|_\infty &= \mathcal{O}\left(\frac{1}{n}\right), & \left\|\vec{t}\right\|_1 &= \mathcal{O}(n), \\ \|\vec{u}\|_\infty &= \mathcal{O}(n), & \|\vec{u}\|_1 &= \mathcal{O}\left(n^2\right), \\ \|\vec{v}\|_\infty &= \mathcal{O}(1), & \|\vec{v}\|_1 &= \mathcal{O}\left(n^2\right). \end{aligned} \tag{7.11}$$

Proof. The claims about the asymptotic scaling of $\left\|\vec{t}\right\|_\infty$, $\|\vec{u}\|_\infty$, and $\|\vec{v}\|_\infty$ follow from Lemma 41. We then obtain the scaling of the vector 1-norm from the triangle inequality. □

For the inductive step, we consider a general second-quantized operator of the form

$$W = \sum_{\vec{j},\vec{k},\vec{l}} w_{\vec{j},\vec{k},\vec{l}} \underbrace{\cdots \big(A^\dagger_{j_x} A_{k_x}\big) \cdots (N_{l_y}) \cdots}_{\text{at most } q \text{ operators}} \tag{7.12}$$

where $\vec{j}$, $\vec{k}$, and $\vec{l}$ denote vectors of orbitals, with total length at most q. We keep track of the number of $A^\dagger_{j_x} A_{k_x}$ and N_{l_y} in each summand; the largest such number q is called the "layer" of W. We compute the commutator between the kinetic/potential operator and a general second-quantized operator in Proposition 44.

Lemma 43 (Commutation rules of second-quantized operators)**.** *The following commutation rules hold for second-quantized operators:*

$$\begin{aligned}
\big[A^\dagger_j A_k, A^\dagger_l A_m\big] &= \delta_{kl} A^\dagger_j A_m - \delta_{jm} A^\dagger_l A_k, \\
\big[A^\dagger_j A_k, N_l\big] &= \delta_{kl} A^\dagger_j A_l - \delta_{jl} A^\dagger_l A_k, \\
\big[A^\dagger_j A_k, N_l N_m\big] &= \big(\delta_{kl} A^\dagger_j A_l - \delta_{jl} A^\dagger_l A_k\big) N_m + N_l \big(\delta_{km} A^\dagger_j A_m - \delta_{jm} A^\dagger_m A_k\big),
\end{aligned} \tag{7.13}$$

where δ_{kl} is the Kronecker-delta function.

Proof. The first rule is proved by [57, (1.8.14)]. The other rules follow from the definition of the number operator $N_l = A^\dagger_l A_l$ and the commutation relation $\big[AB, C\big] = A\big[B, C\big] + \big[A, C\big]B$ for any operators A, B, and C.

□

Proposition 44. *Let an electronic structure Hamiltonian be given as in (7.4).*

The following statements hold for a general second-quantized operator W with q layers:

1. $\widetilde{W} = [T, W]$ *is an operator with* q *layers and* $\left\|\vec{\widetilde{w}}\right\|_1 \leq 2qn \left\|\vec{t}\right\|_\infty \left\|\vec{w}\right\|_1$;
2. $\widetilde{W} = [U, W]$ *is an operator with* q *layers and* $\left\|\vec{\widetilde{w}}\right\|_1 \leq 2q \left\|\vec{u}\right\|_\infty \left\|\vec{w}\right\|_1$; *and*
3. $\widetilde{W} = [V, W]$ *is an operator with* $q+1$ *layers and* $\left\|\vec{\widetilde{w}}\right\|_1 \leq 4qn \left\|\vec{v}\right\|_\infty \left\|\vec{w}\right\|_1$.

Proof. For Statement 1, we have

$$\begin{aligned} \widetilde{W} = [T, W] &= \left[\sum_{\alpha,\beta} t_{\alpha,\beta} A_\alpha^\dagger A_\beta, \sum_{\vec{j},\vec{k},\vec{l}} w_{\vec{j},\vec{k},\vec{l}} \cdots \left(A_{j_x}^\dagger A_{k_x}\right) \cdots (N_{l_y}) \cdots \right] \\ &= \sum_{\alpha,\beta} \sum_{\vec{j},\vec{k},\vec{l}} t_{\alpha,\beta} w_{\vec{j},\vec{k},\vec{l}} \Big[A_\alpha^\dagger A_\beta, \ \cdots \left(A_{j_x}^\dagger A_{k_x}\right) \cdots (N_{l_y}) \cdots \Big]. \end{aligned} \tag{7.14}$$

Performing the commutation sequentially, it suffices to consider

$$\begin{aligned} &\cdots \left[A_\alpha^\dagger A_\beta, A_{j_x}^\dagger A_{k_x} \right] \cdots (N_{l_y}) \cdots \\ &\cdots \left(A_{j_x}^\dagger A_{k_x}\right) \cdots \left[A_\alpha^\dagger A_\beta, N_{l_y} \right] \cdots \end{aligned} \tag{7.15}$$

For fixed α, β, $\vec{j}$, $\vec{k}$, $\vec{l}$, there are at most q such commutators.

For the first type of commutator, we have from Lemma 43 that

$$\left[A_\alpha^\dagger A_\beta, A_{j_x}^\dagger A_{k_x}\right] = \delta_{\beta,j_x} A_\alpha^\dagger A_{k_x} - \delta_{\alpha,k_x} A_{j_x}^\dagger A_\beta. \tag{7.16}$$

Without loss of generality, consider the first term; its contribution to $\left\|\vec{\widetilde{w}}\right\|_1$ is at

most

$$\sum_{\alpha,\beta}\sum_{\vec{j},\vec{k},\vec{l}}\delta_{\beta,j_x}\left|t_{\alpha,\beta}w_{\vec{j},\vec{k},\vec{l}}\right|=\sum_{\alpha,\vec{j},\vec{k},\vec{l}}\left|t_{\alpha,j_x}w_{\vec{j},\vec{k},\vec{l}}\right|\leq n\left\|\vec{t}\right\|_{\infty}\left\|\vec{w}\right\|_{1}. \tag{7.17}$$

Similarly, we use Lemma 43 to analyze the second type of commutator

$$\left[A_{\alpha}^{\dagger}A_{\beta},N_{l_y}\right]=\delta_{l_y,\beta}A_{\alpha}^{\dagger}A_{\beta}-\delta_{l_y,\alpha}A_{\alpha}^{\dagger}A_{\beta} \tag{7.18}$$

and find its contribution to $\left\|\vec{\widetilde{w}}\right\|_1$ as

$$\sum_{\alpha,\beta}\sum_{\vec{j},\vec{k},\vec{l}}\delta_{l_y,\beta}\left|t_{\alpha,\beta}w_{\vec{j},\vec{k},\vec{l}}\right|=\sum_{\alpha,\vec{j},\vec{k},\vec{l}}\left|t_{\alpha,l_y}w_{\vec{j},\vec{k},\vec{l}}\right|\leq n\left\|\vec{t}\right\|_{\infty}\left\|\vec{w}\right\|_{1}. \tag{7.19}$$

For Statement 2, we have

$$\begin{aligned}\widetilde{W}=\left[U,W\right]&=\left[\sum_{\alpha}u_{\alpha}N_{\alpha},\sum_{\vec{j},\vec{k},\vec{l}}w_{\vec{j},\vec{k},\vec{l}}\cdots\left(A_{j_x}^{\dagger}A_{j_x}\right)\cdots\left(N_{l_y}\right)\cdots\right]\\&=\sum_{\alpha}\sum_{\vec{j},\vec{k},\vec{l}}u_{\alpha}w_{\vec{j},\vec{k},\vec{l}}\left[N_{\alpha},\ \cdots\left(A_{j_x}^{\dagger}A_{k_x}\right)\cdots\left(N_{l_y}\right)\cdots\right].\end{aligned} \tag{7.20}$$

Performing the commutation sequentially, it suffices to consider

$$\cdots\left[N_{\alpha},A_{j_x}^{\dagger}A_{k_x}\right]\cdots\left(N_{l_y}\right)\cdots \tag{7.21}$$

For fixed α, $\vec{j}$, $\vec{k}$, $\vec{l}$, there are at most q such commutators. We use Lemma 43 again to get

$$\left[N_{\alpha},A_{j_x}^{\dagger}A_{k_x}\right]=\delta_{\alpha,j_x}A_{j_x}^{\dagger}A_{k_x}-\delta_{\alpha,k_x}A_{j_x}^{\dagger}A_{k_x} \tag{7.22}$$

and find its contribution to $\left\|\vec{\widetilde{w}}\right\|_1$ as

$$\sum_{\alpha}\sum_{\vec{j},\vec{k},\vec{l}}\delta_{\alpha,j_x}\left|u_\alpha w_{\vec{j},\vec{k},\vec{l}}\right| = \sum_{\vec{j},\vec{k},\vec{l}}\left|u_{j_x} w_{\vec{j},\vec{k},\vec{l}}\right| \leq \|\vec{u}\|_\infty \|\vec{w}\|_1 . \tag{7.23}$$

For Statement 3, we have

$$\begin{aligned}\widetilde{W} = [V,W] &= \left[\sum_{\alpha,\beta} v_{\alpha,\beta}N_\alpha N_\beta, \sum_{\vec{j},\vec{k},\vec{l}} w_{\vec{j},\vec{k},\vec{l}}\cdots\left(A^\dagger_{j_x}A_{k_x}\right)\cdots(N_{l_y})\cdots\right] \\ &= \sum_{\alpha,\beta}\sum_{\vec{j},\vec{k},\vec{l}} v_{\alpha,\beta}w_{\vec{j},\vec{k},\vec{l}}\left[N_\alpha N_\beta,\ \cdots\left(A^\dagger_{j_x}A_{k_x}\right)\cdots(N_{l_y})\cdots\right].\end{aligned} \tag{7.24}$$

Performing the commutation sequentially, it suffices to consider

$$\cdots\left[N_\alpha N_\beta, A^\dagger_{j_x}A_{k_x}\right]\cdots(N_{l_y})\cdots \tag{7.25}$$

For fixed α, β, $\vec{j}$, $\vec{k}$, $\vec{l}$, there are at most q such commutators. Using Lemma 43, we have

$$\left[N_\alpha N_\beta, A^\dagger_{j_x}A_{k_x}\right] = \left(\delta_{\alpha,j_x}A^\dagger_{j_x}A_{k_x} - \delta_{\alpha,k_x}A^\dagger_{j_x}A_{k_x}\right)N_\beta + N_\alpha\left(\delta_{\beta,j_x}A^\dagger_{j_x}A_{k_x} - \delta_{\beta,k_x}A^\dagger_{j_x}A_{k_x}\right). \tag{7.26}$$

Without loss of generality, consider the first term; its contribution to $\left\|\vec{\widetilde{w}}\right\|_1$ is at most

$$\sum_{\alpha,\beta}\sum_{\vec{j},\vec{k},\vec{l}}\delta_{\alpha,j_x}\left|v_{\alpha,\beta}w_{\vec{j},\vec{k},\vec{l}}\right| = \sum_{\beta,\vec{j},\vec{k},\vec{l}}\left|v_{j_x,\beta}w_{\vec{j},\vec{k},\vec{l}}\right| \leq n\|\vec{v}\|_\infty \|\vec{w}\|_1 . \tag{7.27}$$

□

We now compute the scaling of the spectral norm of

$$W = \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right], \tag{7.28}$$

by induction, where $H_\gamma \in \{T, U, V\}$. In the base case where $p = 1$, we have from Proposition 42 and Proposition 44 that the coefficients of W have 1-norm in $\mathcal{O}\left(n^2\right)$, which implies $\|W\| = \mathcal{O}\left(n^2\right)$. For the inductive step, suppose that $W = \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right]$ is a second-quantized operator whose coefficients have vector one-norm in $\mathcal{O}\left(n^p\right)$. Then Proposition 44 implies that $\left[T, W\right]$, $\left[U, W\right]$, and $\left[V, W\right]$ are second-quantized operators and their coefficients have 1-norm in $\mathcal{O}\left(n^{p+1}\right)$. This proves that

$$\widetilde{\alpha}_{\text{comm}} = \sum_{\gamma_1, \gamma_2, \ldots, \gamma_{p+1}} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right] \right\| = \mathcal{O}\left(n^{p+1}\right). \tag{7.29}$$

Theorem 39 and Corollary 40 then imply a Trotter number of $r = \mathcal{O}\left((nt)^{1+1/p}/\epsilon^{1/p}\right)$ suffices to simulate with accuracy ϵ. Choosing p sufficiently large, letting ϵ be constant, and implementing each Trotter step as in [46, 76], we have the gate complexity

$$n^{2+o(1)} t^{1+o(1)} \tag{7.30}$$

for simulating plane-wave electronic structure in second quantization.

7.3 k-local Hamiltonians

A Hamiltonian is k-local if it can be expressed as a linear combination of terms, each of which acts nontrivially on at most $k = \mathcal{O}(1)$ qubits. Such Hamiltonians, especially 2-local ones, are ubiquitous in physics. The first explicit quantum simulation algorithm by Lloyd was specifically developed for simulating k-local Hamiltonians [70] and later work provided more advanced approaches based on the linear-combination-of-unitary technique [13–15, 73, 74, 76]. Here, we give an improved product-formula algorithm that can be advantageous over previous simulation methods.

We consider a k-local Hamiltonian acting on n qubits

$$H = \sum_{j_1,\ldots,j_k} H_{j_1,\ldots,j_k}, \tag{7.31}$$

where each $H_{j_1,\ldots,j_k}$ acts nontrivially only on qubits $j_1, \ldots, j_k$. We say $H_{j_1,\ldots,j_k}$ has support $\{j_1, \ldots, j_k\}$, denoting

$$\mathcal{S}\big(H_{j_1,\ldots,j_k}\big) := \{j_1, \ldots, j_k\}. \tag{7.32}$$

We may assume that the summands are unitaries up to scaling and can be implemented with constant cost, for otherwise we expand them further with respect to the Pauli operators. The fastest previous approach to simulating a general k-local

Hamiltonian is the qubitization algorithm by Low and Chuang [74], which has gate complexity $\widetilde{\mathcal{O}}\left(n^k \left\|H\right\|_1 t\right)$ where $\left\|H\right\|_1 = \sum_{j_1,\ldots,j_k} \left\|H_{j_1,\ldots,j_k}\right\|$.

To compare with the product-formula algorithm, we need to analyze the nested commutators $[H_{\gamma_{p+1}}, \cdots [H_{\gamma_2}, H_{\gamma_1}]]$, where each H_γ is some local operator $H_{j_1,\ldots,j_k}$. In order for this commutator to be nonzero, every operator must have support that overlaps with the support of operators from the inner layers. Specifically, we claim that the operator

$$W_{\gamma_1,\ldots,\gamma_{p+1}} \equiv \left[H_{\gamma_{p+1}}, \ldots, [H_{\gamma_2}, H_{\gamma_1}]\right] \tag{7.33}$$

is supported on at most $k + p(k-1)$ qubits and $\sum_{\gamma_1,\ldots,\gamma_{p+1}=1}^{\Gamma} \left\|W_{\gamma_1,\ldots,\gamma_{p+1}}\right\| = \mathcal{O}\left(\left|\left|\left|H\right|\right|\right|_1^p \left\|H\right\|_1\right)$, where we have used the 1-norm $\left\|H\right\|_1 = \sum_{j_1,\ldots,j_k} \left\|H_{j_1,\ldots,j_k}\right\|$ and the induced 1-norm $\left|\left|\left|H\right|\right|\right|_1 = \max_l \max_{j_l} \sum_{j_1,\ldots,j_{l-1},j_{l+1},\ldots,j_k} \left\|H_{j_1,\ldots,j_k}\right\|$. We prove this claim by induction on p. For $p = 1$, the commutator $W_{\gamma_1,\gamma_2} = [H_{\gamma_2}, H_{\gamma_1}]$ takes the form $\left[H_{j_1,\ldots,j_k}, H_{i_1,\ldots,i_k}\right]$, which is nonzero only when there exist $l, m = 1, \ldots, k$ such that $j_l = i_m$. It then follows that W_{γ_1,γ_2} is supported on at most $2k-1$ qubits and that

$$\begin{aligned}\sum_{\substack{j_1,\ldots,j_k,\\ i_1,\ldots,i_k}} \left\|[H_{j_1,\ldots,j_k}, H_{i_1,\ldots,i_k}]\right\| \leq & 2k^2 \max_l \max_{j_l} \sum_{\substack{j_1,\ldots,j_{l-1},\\ j_{l+1},\ldots,j_k}} \left\|H_{j_1,\ldots,j_k}\right\| \sum_{i_1,\ldots,i_k} \left\|H_{i_1,\ldots,i_k}\right\| \\ = & \mathcal{O}\left(\left|\left|\left|H\right|\right|\right|_1 \left\|H\right\|_1\right),\end{aligned} \tag{7.34}$$

which proves the claim for $p = 1$.

Suppose that the claim holds up to $p-1$. Following a similar argument, we have

$$\begin{aligned}
&\sum_{\gamma_1,\ldots,\gamma_{p+1}=1}^{\Gamma} \left\|W_{\gamma_1,\ldots,\gamma_{p+1}}\right\| \\
=&\sum_{j_1,\ldots,j_k} \sum_{\gamma_1,\ldots,\gamma_p=1}^{\Gamma} \left\|\left[H_{j_1,\ldots,j_k}, W_{\gamma_1,\ldots,\gamma_p}\right]\right\| \\
\leq& 2k\big(k+(p-1)(k-1)\big) \max_l \max_{j_l} \sum_{\substack{j_1,\ldots,j_{l-1},\\ j_{l+1},\ldots,j_k}} \left\|H_{j_1,\ldots,j_k}\right\| \sum_{\gamma_1,\ldots,\gamma_p=1}^{\Gamma} \left\|W_{\gamma_1,\ldots,\gamma_p}\right\| \\
=& 2k\big(k+(p-1)(k-1)\big) \left|\!\left|\!\left|H\right|\!\right|\!\right|_1 \cdot \mathcal{O}\left(\left|\!\left|\!\left|H\right|\!\right|\!\right|_1^{p-1} \|H\|_1\right) = \mathcal{O}\left(\left|\!\left|\!\left|H\right|\!\right|\!\right|_1^{p} \|H\|_1\right).
\end{aligned} \tag{7.35}$$

Since the support of $H_{j_1,\ldots,j_k}$ and $W_{\gamma_1,\ldots,\gamma_p}$ overlaps, the operator $W_{\gamma_1,\ldots,\gamma_{p+1}}$ acts nontrivially on at most $k+p(k-1)$ qubits. This completes the induction.

Theorem 39 and Corollary 40 then imply that a Trotter number of $r = \mathcal{O}\left(\left|\!\left|\!\left|H\right|\!\right|\!\right|_1 \|H\|_1^{1/p} t^{1+1/p}/\epsilon^{1/p}\right)$ suffices to simulate with accuracy ϵ. Choosing p sufficiently large, letting ϵ be constant, and implementing each Trotter step with $\Theta\left(n^k\right)$ gates, we have the total gate complexity

$$n^k \left|\!\left|\!\left|H\right|\!\right|\!\right|_1 \|H\|_1^{o(1)} t^{1+o(1)} \tag{7.36}$$

for simulating a k-local Hamiltonian H.

We know from Section 2.1 that the norm inequality $\left|\!\left|\!\left|H\right|\!\right|\!\right|_1 \leq \|H\|_1$ always holds. In fact, the gap between these two norms can be significant for many k-local Hamiltonians. As an example, we consider n-qubit power-law interactions

$H = \sum_{\vec{i},\vec{j}\in\Lambda} H_{\vec{i},\vec{j}}$ with exponent α [102], where $\Lambda \subseteq \mathbb{R}^d$ is a d-dimensional square lattice, $H_{\vec{i},\vec{j}}$ is an operator supported on two sites $\vec{i},\vec{j} \in \Lambda$, and

$$\left\| H_{\vec{i},\vec{j}} \right\| \leq \begin{cases} 1, & \text{if } \vec{i} = \vec{j}, \\ \frac{1}{\left\| \vec{i}-\vec{j} \right\|_2^{\alpha}}, & \text{if } \vec{i} \neq \vec{j}. \end{cases} \tag{7.37}$$

Examples of such systems include those that interact via the Coulomb interactions ($\alpha = 1$), the dipole-dipole interactions ($\alpha = 3$), and the van der Waals interactions ($\alpha = 6$).

To analyze the performance of product formulas, we use the following lemma, whose proof can be found in [37, 102].

Lemma 45. *Given an n-qubit d-dimensional square lattice $\Lambda \subseteq \mathbb{R}^d$, it holds*

$$\sum_{\vec{j}\in\Lambda\setminus\{\vec{0}\}} \frac{1}{\left\| \vec{j} \right\|_2^{\alpha}} = \begin{cases} \mathcal{O}\left(n^{1-\alpha/d}\right), & \text{for } 0 \leq \alpha < d, \\ \mathcal{O}\left(\log n\right), & \text{for } \alpha = d, \\ \mathcal{O}\left(1\right), & \text{for } \alpha > d. \end{cases} \tag{7.38}$$

Furthermore, for $\alpha > d$ and $x > 0$, we have

$$\sum_{\vec{j}\in\Lambda, \left\| \vec{j} \right\|_2 \geq x} \frac{1}{\left\| \vec{j} \right\|_2^{\alpha}} = \mathcal{O}\left(\frac{1}{x^{\alpha-d}}\right). \tag{7.39}$$

Given a power-law Hamiltonian H with exponent α, we use Lemma 45 to

compute the scaling of its induced 1-norm

$$|||H|||_1 \leq \max_{\vec{i}} \sum_{\vec{j}\neq\vec{i}} \left(1 + \frac{1}{\left\|\vec{i}-\vec{j}\right\|_2^{\alpha}}\right) = \begin{cases} \mathcal{O}\left(n^{1-\alpha/d}\right), & \text{for } 0 \leq \alpha < d, \\ \mathcal{O}\left(\log n\right), & \text{for } \alpha = d, \\ \mathcal{O}\left(1\right), & \text{for } \alpha > d, \end{cases} \tag{7.40}$$

and 1-norm

$$\|H\|_1 \leq \sum_{\vec{i}} \sum_{\vec{j}\neq\vec{i}} \left(1 + \frac{1}{\left\|\vec{i}-\vec{j}\right\|_2^{\alpha}}\right) = \begin{cases} \mathcal{O}\left(n^{2-\alpha/d}\right), & \text{for } 0 \leq \alpha < d, \\ \mathcal{O}\left(n\log n\right), & \text{for } \alpha = d, \\ \mathcal{O}\left(n\right), & \text{for } \alpha > d. \end{cases} \tag{7.41}$$

Thus the product-formula algorithm has gate complexity

$$g_\alpha = \begin{cases} n^{3-\frac{\alpha}{d}+o(1)}t^{1+o(1)} & \text{for } 0 \leq \alpha < d, \\ n^{2+o(1)}t^{1+o(1)} & \text{for } \alpha \geq d, \end{cases} \tag{7.42}$$

which has better n-dependence than the qubitization approach [74].

7.4 Rapidly decaying power-law interactions

We now consider d-dimensional power-law interactions $1/x^\alpha$ with exponent $\alpha > 2d$ and interactions that decay exponentially with distance. Although these

Hamiltonians can be simulated using algorithms for k-local Hamiltonians, more efficient methods exist that exploit the locality of the systems [102]. We show that product formulas can also leverage locality to provide an even faster simulation.

We first consider an n-qubit d-dimensional power-law Hamiltonian $H = \sum_{i,j\in\Lambda}$

$H_{\vec{i},\vec{j}}$ with exponent $\alpha > 2d$. Such a Hamiltonian represents a rapidly decaying long-range system that becomes nearest-neighbor interacting in the limit $\alpha \to \infty$. For $\alpha > 2d$, the state-of-the-art simulation algorithm decomposes the evolution based on the Lieb-Robinson bound with gate complexity $\widetilde{\mathcal{O}}\left((nt)^{1+2d/(\alpha-d)}\right)$ [102]. We give an improved approach using product formulas which has gate complexity $(nt)^{1+d/(\alpha-d)+o(1)}$.

The idea of our approach is to simulate a truncated Hamiltonian $\widetilde{H} = \sum_{\|\vec{i}-\vec{j}\|_2\leq\ell}$

$H_{\vec{i},\vec{j}}$ by taking only the terms $H_{\vec{i},\vec{j}}$ where $\|\vec{i}-\vec{j}\|_2$ is not more than ℓ, a parameter that we determine later. The resulting $\widetilde{H}$ is a 2-local Hamiltonian with 1-norm $\|\widetilde{H}\|_1 = \mathcal{O}(n)$ and induced 1-norm $\|\|\widetilde{H}\|\|_1 = \mathcal{O}(1)$. Theorem 39 and Corollary 40 then imply that a Trotter number of $r = \mathcal{O}\left(n^{1/p}t^{1+1/p}/\epsilon^{1/p}\right)$ suffices to simulate with accuracy ϵ. Choosing p sufficiently large, letting ϵ be constant, and implementing each Trotter step with $\mathcal{O}\left(n\ell^d\right)$ gates, we have the total gate complexity $\ell^d(nt)^{1+o(1)}$ for simulating $\widetilde{H}$.

We know from Corollary 5 that the approximation of $\exp(-iHt)$ by $\exp(-i\widetilde{H}t)$

has error

$$\left\| e^{-itH} - e^{-it\widetilde{H}} \right\| = \mathcal{O}\left(\left\| H - \widetilde{H} \right\| t \right), \tag{7.43}$$

where $\|H - \widetilde{H}\| = \mathcal{O}\left(n/\ell^{\alpha-d}\right)$ for all $\alpha > 2d$. To make this at most $\mathcal{O}(\epsilon)$, we choose the cut-off $\ell = \Theta\left((nt/\epsilon)^{1/(\alpha-d)}\right)$. Note that we require $nt \geq \epsilon$ and $t \leq \epsilon n^{\alpha/d-2}$ so that $n^{1/d} \geq \ell \geq 1$. This implies the gate complexity

$$(nt)^{1+d/(\alpha-d)+o(1)}, \tag{7.44}$$

which is better than the state-of-the-art algorithm based on Lieb-Robinson bounds [102].

We also consider interactions that decay exponentially with the distance x as $e^{-\beta x}$:

$$\left\| H_{\vec{i},\vec{j}} \right\| \leq e^{-\beta \left\| \vec{i} - \vec{j} \right\|_2}, \tag{7.45}$$

where $\beta > 0$ is a constant. Although such interactions are technically long-range, their fast decay makes them quasi-local for most applications in physics. Our approach to simulating such a quasi-local system is similar to that for the rapidly decaying power-law Hamiltonian, except we choose the cut-off $\ell = \Theta(\log(nt/\epsilon))$, giving a product-formula algorithm with gate complexity

$$(nt)^{1+o(1)}. \tag{7.46}$$

Our result for quasi-local systems is asymptotically the same as a recent result for nearest-neighbor Hamiltonians [36]. For rapidly decaying power-law systems, we reproduce the nearest-neighbor case [36] in the limit $\alpha \to \infty$.

7.5 Clustered Hamiltonians

We now consider the application of our theory to simulating clustered Hamiltonians [86]. Such systems appear naturally in the study of classical fragmentation methods and quantum mechanics/molecular mechanics methods for simulating large molecules. Peng, Harrow, Ozols, and Wu recently proposed a hybrid simulator for clustered Hamiltonians [86]. Here, we show that the performance of their simulator can be significantly improved using our Trotter error bound.

Let H be a Hamiltonian acting on n qubits. Following the same setting as in [86], we assume that each term in H acts on at most two qubits with spectral norm at most one, and each qubit interacts with at most a constant number d' of other qubits. We further assume that the qubits are grouped into multiple parties and write

$$H = A + B = \sum_l H_l^{(1)} + \sum_l H_l^{(2)}, \quad \forall l : \left\| H_l^{(1)} \right\|, \left\| H_l^{(2)} \right\| \leq 1, \tag{7.47}$$

where terms in A act on qubits within a single party and terms in B act between two different parties.

The key step in the approach of Peng et al. is to group the terms within each party in A and simulate the resulting Hamiltonian. This is accomplished by applying product formulas to the decomposition

$$H = A + \sum_l H_l^{(2)}. \tag{7.48}$$

Using the first-order Lie-Trotter formula, Reference [86] chooses the Trotter number

$$r = \mathcal{O}\left(\frac{h_B^2 t^2}{\epsilon}\right) \tag{7.49}$$

to ensure that the error of the decomposition is at most ϵ, where $h_B = \sum_l \|H_l^{(2)}\|$ is the interaction strength. Here, we show that it suffices to take

$$r = \mathcal{O}\left(\frac{d'^{\frac{1+p}{2}} h_B^{\frac{1}{p}} t^{1+\frac{1}{p}}}{\epsilon^{\frac{1}{p}}}\right) = \mathcal{O}\left(\frac{h_B^{1/p} t^{1+1/p}}{\epsilon^{1/p}}\right) \tag{7.50}$$

using a pth-order product formula

$$\mathscr{S}(t) = e^{-ita_\Upsilon A} \prod_l e^{-ita_{(\Upsilon,l)} H_l^{(2)}} \cdots e^{-ita_1 A} \prod_l e^{-ita_{(1,l)} H_l^{(2)}}. \tag{7.51}$$

This improves the analysis of [86] for the first-order formula and extends the result to higher-order cases.

In light of Theorem 39 and Corollary 40, we need to compute

$$\sum_{\gamma_1,\gamma_2,\ldots,\gamma_{p+1}} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1} \right] \cdots \right] \right\|, \tag{7.52}$$

where each H_γ is either $H_l^{(2)}$ or A. Since $\left[A, A\right] = 0$ and $\left[H_\gamma, A\right] = -\left[A, H_\gamma\right]$, we may without loss of generality assume that $H_{\gamma_1} = H_{l_1}^{(2)}$, i.e.,

$$\sum_{\gamma_1,\gamma_2,\ldots,\gamma_{p+1}} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1} \right] \cdots \right] \right\| = \sum_{l_1,\gamma_2,\ldots,\gamma_{p+1}} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{l_1}^{(2)} \right] \cdots \right] \right\|. \tag{7.53}$$

We now replace each A by $\sum_l H_l^{(1)}$ and apply the triangle inequality to get

$$\sum_{l_1,\gamma_2,\ldots,\gamma_{p+1}} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{l_1}^{(2)} \right] \cdots \right] \right\| \leq \sum_{l_1,l_2,\ldots,l_{p+1}} \left\| \left[K_{l_{p+1}}, \cdots \left[K_{l_2}, H_{l_1}^{(2)} \right] \cdots \right] \right\|, \tag{7.54}$$

where each K_l is either $H_l^{(1)}$ or $H_l^{(2)}$. Since each qubit supports at most d' terms and each term acts on at most two qubits,

$$\sum_{l_1,l_2,\ldots,l_{p+1}} \left\| \left[K_{l_{p+1}}, \cdots \left[K_{l_2}, H_{l_1}^{(2)} \right] \cdots \right] \right\| = \mathcal{O}\left(d'^p \cdots d'^2 d' \sum_{l_1} \left\| H_{l_1}^{(2)} \right\| \right) = \mathcal{O}\left(d'^{\frac{(1+p)p}{2}} h_B \right). \tag{7.55}$$

This completes the proof.

The hybrid simulator of [86] has runtime $2^{\mathcal{O}(r \cdot \mathrm{cc}(g))}$, where r is the Trotter number and $\mathrm{cc}(g)$ is the contraction complexity of the interaction graph g between the parties. Our improved choice of r thus provides a dramatic improvement.

7.6 Numerics

We have analyzed the error of higher-order product formulas in Section 6.5. That analysis is sufficient to establish the commutator scaling in Theorem 39, but the resulting bounds have large prefactors. Here, we propose heuristic strategies to tighten the analysis and numerically benchmark our bounds for nearest-neighbor lattice Hamiltonians. Throughout this section, we assume H is Hermitian, $t \in \mathbb{R}$, and consider the real-time evolution e^{-itH}.

We first consider a Hamiltonian $H = A + B$ consisting of two summands. The ideal evolution under H for time t is e^{-itH}, which we decompose using the fourth-order product formula $\mathscr{S}_4(t)$. Recall from (2.14) that $\mathscr{S}_4(t)$ is defined by

$$\begin{aligned} \mathscr{S}_2(t) &:= e^{-i\frac{t}{2}A} e^{-itB} e^{-i\frac{t}{2}A}, \\ \mathscr{S}_4(t) &:= \left[\mathscr{S}_2(u_2 t)\right]^2 \mathscr{S}_2((1-4u_2)t) \left[\mathscr{S}_2(u_2 t)\right]^2, \end{aligned} \tag{7.56}$$

with $u_2 := 1/(4 - 4^{1/3})$. Expanding this definition, we obtain

$$\mathscr{S}_4(t) = e^{-ita_6 A} e^{-itb_5 B} e^{-ita_5 A} e^{-itb_4 B} e^{-ita_4 A} e^{-itb_3 B} e^{-ita_3 A} e^{-itb_2 B} e^{-ita_2 A} e^{-itb_1 B} e^{-ita_1 A}, \tag{7.57}$$

where $a_1 := a_6 := \frac{u_2}{2}$, $b_1 := a_2 := b_2 := b_4 := a_5 := b_5 := u_2$, $a_3 := a_4 := \frac{1-3u_2}{2}$, and $b_3 := 1 - 4u_2$.

Without loss of generality, we analyze the additive Trotter error of $\mathscr{S}_4(t)$.

We gave an analysis in Section 6.3 that works for a general product formula, and we improve that here to obtain an error bound for $\mathscr{S}_4(t)$ with small prefactor. To this end, we compute

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}_4(t) - (-iH)\mathscr{S}_4(t) \\
=&\left[e^{-ita_6A}, -ib_5B\right]e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A} \\
&+\left[e^{-ita_6A}e^{-itb_5B}, -ia_5A\right]e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A} \\
&+\cdots \\
&+\left[e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}, -ib_1B\right]e^{-itb_1B}e^{-ita_1A} \\
&+\left[e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}, -ia_1A\right]e^{-ita_1A}.
\end{aligned} \tag{7.58}$$

Performing the commutation sequentially, we have

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}_4(t) - (-iH)\mathscr{S}_4(t) \\
=&\left[e^{-ita_6A}, -ib_5B\right]e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A} \\
&+e^{-ita_6A}\left[e^{-itb_5B}, -ia_5A\right]e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A} \\
&+\cdots
\end{aligned} \tag{7.59}$$

$$
\begin{aligned}
&+ e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}\left[e^{-ita_2A},-ib_1B\right]e^{-itb_1B}e^{-ita_1A}\\
&+ e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}\left[e^{-ita_3A},-ib_1B\right]e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}\\
&+ e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}\left[e^{-ita_4A},-ib_1B\right]e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}\\
&+ e^{-ita_6A}e^{-itb_5B}\left[e^{-ita_5A},-ib_1B\right]e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}\\
&+ \left[e^{-ita_6A},-ib_1B\right]e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}\\
&+ e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}\left[e^{-itb_1B},-ia_1A\right]e^{-ita_1A}\\
&+ e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}\left[e^{-itb_2B},-ia_1A\right]e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}\\
&+ e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}\left[e^{-itb_3B},-ia_1A\right]e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}\\
&+ e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}\left[e^{-itb_4B},-ia_1A\right]e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}\\
&+ e^{-ita_6A}\left[e^{-itb_5B},-ia_1A\right]e^{-ita_5A}e^{-itb_4B}e^{-ita_4A}e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}.
\end{aligned}
\tag{7.60}
$$

We further define

$$
\begin{aligned}
c_1 &:= a_1, & d_1 &:= b_1,\\
c_2 &:= a_1 + a_2, & d_2 &:= b_1 + b_2,\\
c_3 &:= a_1 + a_2 + a_3, & d_3 &:= b_1 + b_2 + b_3,\\
c_4 &:= a_1 + a_2 + a_3 + a_4, & d_4 &:= b_1 + b_2 + b_3 + b_4,\\
c_5 &:= a_1 + a_2 + a_3 + a_4 + a_5, & d_5 &:= b_1 + b_2 + b_3 + b_4 + b_5,
\end{aligned}
\tag{7.61}
$$

so that

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}_4(t) - (-iH)\mathscr{S}_4(t) \\
=&\left[e^{-ita_6 A}, -id_5 B\right] e^{-itb_5 B} e^{-ita_5 A} e^{-itb_4 B} e^{-ita_4 A} e^{-itb_3 B} e^{-ita_3 A} e^{-itb_2 B} e^{-ita_2 A} e^{-itb_1 B} e^{-ita_1 A} \\
&+ e^{-ita_6 A}\left[e^{-itb_5 B}, -ic_5 A\right] e^{-ita_5 A} e^{-itb_4 B} e^{-ita_4 A} e^{-itb_3 B} e^{-ita_3 A} e^{-itb_2 B} e^{-ita_2 A} e^{-itb_1 B} e^{-ita_1 A} \\
&+ \cdots \\
&+ e^{-ita_6 A} e^{-itb_5 B} e^{-ita_5 A} e^{-itb_4 B} e^{-ita_4 A} e^{-itb_3 B} e^{-ita_3 A} e^{-itb_2 B}\left[e^{-ita_2 A}, -id_1 B\right] e^{-itb_1 B} e^{-ita_1 A} \\
&+ e^{-ita_6 A} e^{-itb_5 B} e^{-ita_5 A} e^{-itb_4 B} e^{-ita_4 A} e^{-itb_3 B} e^{-ita_3 A} e^{-itb_2 B} e^{-ita_2 A}\left[e^{-itb_1 B}, -ic_1 A\right] e^{-ita_1 A}.
\end{aligned} \tag{7.62}
$$

In Section 6.3, we factor out the operator-valued function $\mathscr{S}_4(t)$ from the left-hand side of the above equation as

$$
\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}_4(t) - (-iH)\mathscr{S}_4(t) = \mathscr{S}_4(t)\mathscr{T}(t). \tag{7.63}
$$

This approach suffices to establish the asymptotic bound in Theorem 39 and Corollary 40. However, the resulting function $\mathscr{T}(t)$ contains unitary conjugations with a large number of conjugating layers, which defeats the goal of establishing tight error bounds. We improve this by simultaneously factoring out $\mathscr{S}_{4,\mathrm{left}}(t)$ from the left-hand side of the equation and $\mathscr{S}_{4,\mathrm{right}}(t)$ from the right-hand side, obtaining

$$
\frac{\mathrm{d}}{\mathrm{d}t}\mathscr{S}_4(t) - (-iH)\mathscr{S}_4(t) = \mathscr{S}_{4,\mathrm{left}}(t)\mathscr{T}_4(t)\mathscr{S}_{4,\mathrm{right}}(t), \tag{7.64}
$$

where

$$\begin{aligned}\mathscr{S}_{\text{left}}(t) &:= e^{-ita_6A}e^{-itb_5B}e^{-ita_5A}e^{-itb_4B}e^{-ita_4A},\\ \mathscr{S}_{\text{right}}(t) &:= e^{-itb_3B}e^{-ita_3A}e^{-itb_2B}e^{-ita_2A}e^{-itb_1B}e^{-ita_1A}.\end{aligned} \tag{7.65}$$

It then remains to analyze $\mathscr{T}_4(t)$.

To this end, we use the fact that

$$\begin{aligned}\big[e^{tX},Y\big] &= e^{tX}\int_0^t \mathrm{d}\tau\ e^{-\tau X}\big[X,Y\big]e^{\tau X}\\ &= \int_0^t \mathrm{d}\tau\ e^{\tau X}\big[X,Y\big]e^{-\tau X}e^{tX},\end{aligned} \tag{7.66}$$

for any $t\in\mathbb{R}$ and operators X, Y. We then have from Lemma 1 that

$$\mathscr{S}_4(t) = e^{-itH} + \int_0^t \mathrm{d}\tau_1\ e^{-i(t-\tau_1)H}\mathscr{S}_{4,\text{left}}(\tau_1)\mathscr{T}_4(\tau_1)\mathscr{S}_{4,\text{right}}(\tau_1), \tag{7.67}$$

where

$$\begin{aligned}&\mathscr{T}_4(\tau_1)\\ &= \int_0^{\tau_1}\mathrm{d}\tau_2\ e^{i\tau_1a_4A}e^{i\tau_1b_4B}e^{i\tau_1a_5A}e^{i\tau_1b_5B}e^{i\tau_2a_6A}\big[-ia_6A,-id_5B\big]e^{-i\tau_2a_6A}e^{-i\tau_1b_5B}e^{-i\tau_1a_5A}e^{-i\tau_1b_4B}e^{-i\tau_1a_4A}\\ &\quad+\int_0^{\tau_1}\mathrm{d}\tau_2\ e^{i\tau_1a_4A}e^{i\tau_1b_4B}e^{i\tau_1a_5A}e^{i\tau_2b_5B}\big[-ib_5B,-ic_5A\big]e^{-i\tau_2b_5B}e^{-i\tau_1a_5A}e^{-i\tau_1b_4B}e^{-i\tau_1a_4A}\\ &\quad+\int_0^{\tau_1}\mathrm{d}\tau_2\ e^{i\tau_1a_4A}e^{i\tau_1b_4B}e^{i\tau_2a_5A}\big[-ia_5A,-id_4B\big]e^{-i\tau_2a_5A}e^{-i\tau_1b_4B}e^{-i\tau_1a_4A}\\ &\quad+\int_0^{\tau_1}\mathrm{d}\tau_2\ e^{i\tau_1a_4A}e^{i\tau_2b_4B}\big[-ib_4B,-ic_4A\big]e^{-i\tau_2b_4B}e^{-i\tau_1a_4A}\\ &\quad+\int_0^{\tau_1}\mathrm{d}\tau_2\ e^{i\tau_2a_4A}\big[-ia_4A,-id_3B\big]e^{-i\tau_2a_4A}\end{aligned} \tag{7.68}$$

$$
\begin{aligned}
&+ \int_0^{\tau_1} \mathrm{d}\tau_2 \; e^{-i\tau_2 b_3 B}\big[-ib_3 B, -ic_3 A\big] e^{i\tau_2 b_3 B} \\
&+ \int_0^{\tau_1} \mathrm{d}\tau_2 \; e^{-i\tau_1 b_3 B} e^{-i\tau_2 a_3 A}\big[-ia_3 A, -id_2 B\big] e^{i\tau_2 a_3 A} e^{i\tau_1 b_3 B} \\
&+ \int_0^{\tau_1} \mathrm{d}\tau_2 \; e^{-i\tau_1 b_3 B} e^{-i\tau_1 a_3 A} e^{-i\tau_2 b_2 B}\big[-ib_2 B, -ic_2 A\big] e^{i\tau_2 b_2 B} e^{i\tau_1 a_3 A} e^{i\tau_1 b_3 B} \\
&+ \int_0^{\tau_1} \mathrm{d}\tau_2 \; e^{-i\tau_1 b_3 B} e^{-i\tau_1 a_3 A} e^{-i\tau_1 b_2 B} e^{-i\tau_2 a_2 A}\big[-ia_2 A, -id_1 B\big] e^{i\tau_2 a_2 A} e^{i\tau_1 b_2 B} e^{i\tau_1 a_3 A} e^{i\tau_1 b_3 B} \\
&+ \int_0^{\tau_1} \mathrm{d}\tau_2 \; e^{-i\tau_1 b_3 B} e^{-i\tau_1 a_3 A} e^{-i\tau_1 b_2 B} e^{-i\tau_1 a_2 A} e^{-i\tau_2 b_1 B}\big[-ib_1 B, -ic_1 A\big] e^{i\tau_2 b_1 B} e^{i\tau_1 a_2 A} e^{i\tau_1 b_2 B} e^{i\tau_1 a_3 A} e^{i\tau_1 b_3 B}.
\end{aligned}
\tag{7.69}
$$

The operator-valued function $\mathscr{T}_4(\tau_1)$ has the order condition $\mathscr{T}_4(\tau_1) = O(\tau_1^4)$, which follows from Proposition 36 and the fact that $\mathscr{S}_4(t) = e^{-itH} + O(t^5)$. For terms in $\mathscr{T}_4(\tau_1)$, we compute the Taylor expansion of each layer of unitary conjugation as in Section 6.5. In light of Lemma 35, we expand the time variables τ_1 and τ_2 to third order, as there already exists the double integral $\int_0^t \mathrm{d}\tau \int_0^{\tau_1} \mathrm{d}\tau_2$. We then apply the triangle inequality to bound the spectral norm of a linear combination of nested commutators of A and B with four nesting layers. Since $[A, A] = [B, B] = 0$ and $[A, B] = [B, A]$, the bound only contains $2^5/4 = 8$ nonzero terms. Altogether,

we obtain

$$\begin{aligned}
&\left\|\mathscr{S}_4(t) - e^{-itH}\right\| \\
\leq& t^5\Big(0.0047\left\|\left[A,\left[A,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0057\left\|\left[A,\left[A,\left[B,\left[B,A\right]\right]\right]\right]\right\| \\
&+ 0.0046\left\|\left[A,\left[B,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0074\left\|\left[A,\left[B,\left[B,\left[B,A\right]\right]\right]\right]\right\| \\
&+ 0.0097\left\|\left[B,\left[A,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0097\left\|\left[B,\left[A,\left[B,\left[B,A\right]\right]\right]\right]\right\| \\
&+ 0.0173\left\|\left[B,\left[B,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0284\left\|\left[B,\left[B,\left[B,\left[B,A\right]\right]\right]\right]\right\|\Big),
\end{aligned} \tag{7.70}$$

assuming $t \geq 0$.

Proposition 46 (Trotter error bound for the fourth-order Suzuki formula with two summands)**.** *Let $H = A + B$ be a Hamiltonian consisting of two summands and $t \geq 0$. Let $\mathscr{S}_4(t)$ be the fourth-order Suzuki formula (2.14). Then,*

$$\begin{aligned}
&\left\|\mathscr{S}_4(t) - e^{-itH}\right\| \\
\leq& t^5\Big(0.0047\left\|\left[A,\left[A,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0057\left\|\left[A,\left[A,\left[B,\left[B,A\right]\right]\right]\right]\right\| \\
&+ 0.0046\left\|\left[A,\left[B,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0074\left\|\left[A,\left[B,\left[B,\left[B,A\right]\right]\right]\right]\right\| \\
&+ 0.0097\left\|\left[B,\left[A,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0097\left\|\left[B,\left[A,\left[B,\left[B,A\right]\right]\right]\right]\right\| \\
&+ 0.0173\left\|\left[B,\left[B,\left[A,\left[B,A\right]\right]\right]\right]\right\| + 0.0284\left\|\left[B,\left[B,\left[B,\left[B,A\right]\right]\right]\right]\right\|\Big).
\end{aligned} \tag{7.71}$$

Although we do not have a rigorous proof of the tightness of our higher-order bounds, numerical evidence suggests that they are close to tight for various systems. We first consider simulating a one-dimensional Heisenberg model with

a random magnetic field (3.1). This system can be simulated to understand the transition between the many-body localized phase and the thermalized phase in condensed matter physics, although a classical simulation is only feasible when the system size is small [78].

We classify the summands of the Hamiltonian into two groups and set

$$\begin{aligned} A &= \sum_{j=1}^{\lfloor \frac{n}{2} \rfloor} \left(X_{2j-1}X_{2j} + Y_{2j-1}Y_{2j} + Z_{2j-1}Z_{2j} + h_{2j-1}Z_{2j-1} \right), \\ B &= \sum_{j=1}^{\lceil \frac{n}{2} \rceil - 1} \left(X_{2j}X_{2j+1} + Y_{2j}Y_{2j+1} + Z_{2j}Z_{2j+1} + h_{2j}Z_{2j} \right). \end{aligned} \tag{7.72}$$

Here, all the summands in A (and B) commute with each other, so we can further decompose exponentials like e^{-ita_kA} (and e^{-itb_kB}) without introducing error, giving a product formula with summands ordered in an *even-odd* pattern [36]. We also consider grouping Hamiltonian summands as

$$H_1 = \sum_{j=1}^{n-1} X_jX_{j+1}, \qquad H_2 = \sum_{j=1}^{n-1} Y_jY_{j+1}, \qquad H_3 = \sum_{j=1}^{n-1} \left(Z_jZ_{j+1} + h_jZ_j \right), \tag{7.73}$$

which we call the X-Y-Z ordering [34]. Similar to the even-odd ordering, the summands in H_1, H_2, and H_3 commute with each other respectively, so the corresponding exponentials can also be decomposed without error. Note that our asymptotic bounds in Theorem 39 and Corollary 40 hold irrespective of the ordering of Hamiltonian summands, but the prefactors will depend on the choice of ordering. Our choice here maximizes the commutativity of the Hamiltonian.

Up to a difference on the boundary condition, Reference [34] estimates the resource requirements of simulating the Heisenberg model using various quantum algorithms. They find that product formulas, especially the fourth-order and the sixth-order formula, can outperform more recent quantum algorithms for simulating small instances of (3.1), although their best Trotter error bound is loose by several orders of magnitude. This is alleviated by [36], which gives a fourth-order bound that overestimates the gate complexity by about a factor of 17. For a fair comparison, we numerically implement our approach to analyze the fourth-order formula $\mathscr{S}_4(t)$ as well (Proposition 46).

For the even-odd ordering, we need to compute all the nested commutators of A and B. We do this by fixing one term $X_{2j-1}X_{2j}+Y_{2j-1}Y_{2j}+Z_{2j-1}Z_{2j}+h_{2j-1}Z_{2j-1}$ of A in the inner-most layer and simplifying all the outer terms using geometrical locality. We then apply the triangle inequality to analyze the summation of terms over $j = 1, \ldots, \lfloor \frac{n}{2} \rfloor$. We use a similar approach to analyze the X-Y-Z ordering. This computes our error bounds for small t. To simulate for a longer time, we divide the evolution into r Trotter steps and apply our bounds within each step. We seek the smallest Trotter number r for which the estimated error is at most some desired ϵ. This can be efficiently computed using a binary search as described in [34].

We compare our improved analysis with the best previous bounds [34, 36] for simulating the Heisenberg model (3.1). Specifically, we consider the so-called

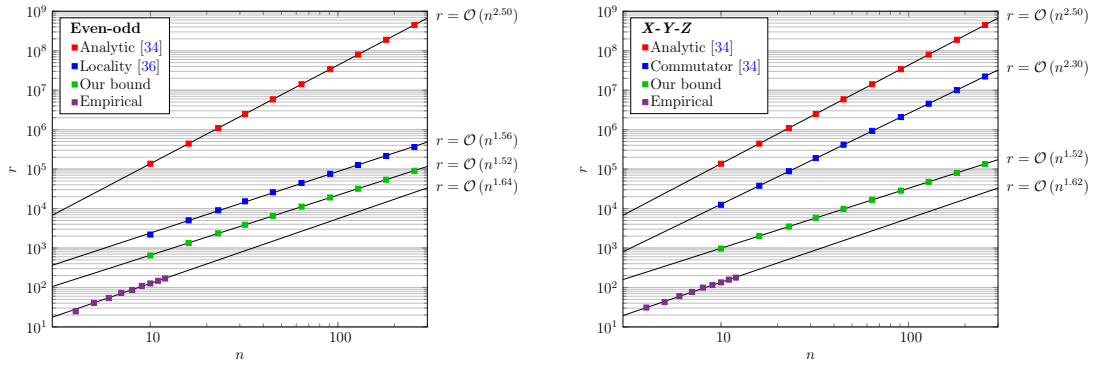

Figure 7.1: Comparison of r for different product-formula bounds for the Heisenberg model. Error bars are omitted as they are negligibly small on the plot. Straight lines show power-law fits to the data. Note that the exponent for the empirical data is based on brute-force simulations of small systems, and thus may not precisely capture the true asymptotic scaling due to finite-size effects.

analytic bound [34], which applies to both the even-odd and the X-Y-Z ordering. The commutator bound of [34] offers a slight improvement over the analytic bound, but its numerical implementation requires extensive classical computations and so we only compare the existing result for the X-Y-Z ordering. Likewise, we compare the locality-based bound of [36] only for the even-odd case, although it can exploit the geometrical locality of the X-Y-Z ordering as well.

To understand how tight our bounds are, we also include the empirical Trotter number by directly computing the error $\left\| \left(\mathscr{S}_4(t/r)\right)^r - e^{-itH} \right\|$ for $n = 4, \ldots, 12$ and extrapolating the results to larger systems. We choose the evolution time $t = n$ and set the simulation accuracy $\epsilon = 10^{-3}$ as in [34] and [36]. For each system size, we generate five instances of Hamiltonians with random coefficients. Our results are plotted in Figure 7.1.

We find that the asymptotic scaling of our new bounds matches that of

the empirical result up to finite-size effects and the prefactors are significantly tightened. At $n = 10$, the Trotter number predicted by our bounds is loose only by a factor of 5.1 for the even-odd ordering of terms and 7.2 for the X-Y-Z ordering. In comparison, the commutator bound of [34] only exploits the commutativity of the lowest-order term of the BCH series and is bottlenecked by the use of tail bounds. The previous bound [36] based on geometrical locality is also uncompetitive since it cannot directly leverage the nested commutators of the Hamiltonian terms.

7.7 Discussion

We have developed a general theory of Trotter error and identified a host of applications to simulating quantum dynamics. We consider Trotter error of various types, including additive error, multiplicative error, and error that appears in the exponent. For each type, we apply the correct order condition to cancel lower-order terms, and represent higher-order ones as explicit nested commutators. Table 7.1 compares our results against the best previous ones for simulating quantum dynamics.

Compared to the analysis of other simulation algorithms such as the truncated Taylor-series algorithm [14] and the qubitization approach [74], the derivation of our Trotter error theory is considerably more involved. However, the resulting error bounds are succinct and easy to evaluate. Theorem 39 shows that Trotter error incurred by decomposing the evolution generated by $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ depends

System	Best previous result	New result
Nearest-neighbor lattice	$(nt)^{1+o(1)}$ (Conjecture), $\widetilde{\mathcal{O}}(nt)$ (Lieb-Robinson bound)	$(nt)^{1+o(1)}$
Electronic structure	$\widetilde{\mathcal{O}}\left(n^2 t\right)$ (Interaction picture)	$n^{2+o(1)}t^{1+o(1)}$
k-local Hamiltonians	$\widetilde{\mathcal{O}}\left(n^k \left\|H\right\|_1 t\right)$ (Qubitization)	$n^k \left\|\!\left\|\!\left\|H\right\|\!\right\|_1 \left\|H\right\|_1^{o(1)} t^{1+o(1)}$
$1/x^\alpha$ $(\alpha < d)$	$\widetilde{O}(n^{4-\alpha/d}t)$ (Qubitization)	$n^{3-\alpha/d+o(1)}t^{1+o(1)}$
$1/x^\alpha$ $(d \leq \alpha \leq 2d)$	$\widetilde{O}(n^3 t)$ (Qubitization)	$n^{2+o(1)}t^{1+o(1)}$
$1/x^\alpha$ $(\alpha > 2d)$	$\widetilde{\mathcal{O}}\left((nt)^{1+2d/(\alpha-d)}\right)$ (Lieb-Robinson bound)	$(nt)^{1+d/(\alpha-d)+o(1)}$
Clustered Hamiltonians	$2^{\mathcal{O}\left(h_B^2 t^2 \operatorname{cc}(g)/\epsilon\right)}$	$2^{\mathcal{O}\left(h_B^{o(1)} t^{1+o(1)} \operatorname{cc}(g)/\epsilon^{o(1)}\right)}$

Table 7.1: Comparison of our results and the best previous results for simulating quantum dynamics.

asymptotically on the quantity $\widetilde{\alpha}_{\text{comm}} = \sum_{\gamma_1,\gamma_2,\ldots,\gamma_{p+1}} \left\| \left[H_{\gamma_{p+1}}, \cdots \left[H_{\gamma_2}, H_{\gamma_1}\right]\right] \right\|$, which can be computed by induction as for nearest-neighbor lattice systems, second-quantized plane-wave electronic structure, k-local Hamiltonians, rapidly decaying interacted systems, and clustered Hamiltonians. We further show how to improve the analysis to find error bounds with small constant prefactors. Numerical simulation suggests that our higher-order error bounds are close to tight for systems with nearest-neighbor interactions, and we hope future work can explore their tightness for other systems.

Our result shows that high-order product formulas can be advantageous for simulating many physical systems. Interestingly, we can often achieve this advantage without using a formula of very large order. For d-dimensional power-law interactions with exponent $\alpha > 2d$, we have shown that the pth-order product-formula algorithm has gate complexity $\mathcal{O}\left((nt)^{1+d/(\alpha-d)+1/p}\right)$, whereas the state-of-

the-art Lieb-Robinson-based approach requires $\widetilde{\mathcal{O}}\left((nt)^{1+2d/(\alpha-d)}\right)$ gates. Product formulas can thus scale better if $p \geq (\alpha - d)/d$, which is small for various physical systems such as the dipole-dipole interactions ($\alpha = 3$) and the Van der Waals interactions ($\alpha = 6$). For other systems such as nearest-neighbor interactions and electronic structure Hamiltonians, product formulas do not exactly match the state-of-the-art result in terms of the asymptotic scaling, but they are still advantageous for simulating systems of small sizes [34, 66].

The complexity of the product-formula approach is determined by both the Trotter number (or Trotter error) and the cost per Trotter step. A naive implementation of each Trotter step exponentiates all the terms in the Hamiltonian, which has a cost that scales with the total number of terms. However, this worst-case complexity can be avoided by truncating the original Hamiltonian, as we have demonstrated in the simulation of rapidly decaying power-law Hamiltonians. Recent studies have proposed other techniques for implementing Trotter steps [3, 66, 67, 107]. Those techniques can be applied in combination with our Trotter error analysis to further speed up the product-formula algorithm.

We have restricted to the evolutions generated by time-independent operators. In the more general case, we have an operator-valued function $\mathscr{H}(\tau) = \sum_{\gamma=1}^{\Gamma} \mathscr{H}_\gamma(\tau)$ and our goal is to simulate the time-ordered evolution $\exp_{\mathcal{T}}\left(\int_0^t \mathrm{d}\tau \sum_{\gamma=1}^{\Gamma} \mathscr{H}_\gamma(\tau)\right)$ [13, 14, 16, 65, 76, 88, 110]. Under certain smoothness assumptions, Reference [110] shows that this evolution can be simulated using product

formulas, although their analysis does not exploit the commutativity of operator summands. We believe our approach can be extended to give improved analysis for time-dependent Hamiltonian simulation, but we leave a detailed study for future work.

Previous work considered several generalized product formulas, such as ones based on the divide-and-conquer construction [54], the randomized construction [35, 84], and the linear-combination-of-unitaries construction [75]. The common underlying idea is to approximate the ideal evolution to pth order using formulas of order q_k, where $q_k \leq p$. Our theory can be applied to represent the q_kth-order Trotter error in terms of nested commutators, thus improving the previous analyses of [35, 54, 75, 84]. This leads to a better understanding of these generalized formulas and justifies their potential utility in quantum simulation.

Several other questions related to our theory deserve further investigation. For example, the spectral-norm error bound computed here would be overly pessimistic if we simulate with a low-energy initial state. It would then be beneficial to change the error metric to the Euclidean distance to avoid the worst-case error propagation. Our analysis has also assumed an operator decomposition $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ given a prior, but one may instead seek an alternative decomposition to maximize the commutativity of operator summands. Finally, we focus on the error analysis within each Trotter step and apply the triangle inequality across different steps, which may be improved upon as hinted in previous work [58, 96].

Chapter 8: Quantum singular value transformation

In this chapter, we develop an algorithmic framework we call "quantum singular value transformation", which is inspired by techniques from quantum simulation. We discuss the core concepts of this framework, including the standard-form encoding, qubitization, and quantum signal processing in Section 8.1, Section 8.2, and Section 8.3, respectively. Quantum singular value transformation unifies a host of existing quantum algorithms and provides a convenient approach to designing new quantum algorithms. We illustrate this in Section 8.4 by applying the framework to implementing principal component regression.

This chapter is partly based on the following paper:

[50] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe, *Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics*, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, pp. 193–204, ACM, 2019, arXiv:1806.01838.

8.1 Standard-form encoding

We consider three vector spaces $\mathcal{H}_1$, $\mathcal{H}_2$ and $\mathcal{H}$ with dimensionality constraint $\dim(\mathcal{H}_1), \dim(\mathcal{H}_2) \leq \dim(\mathcal{H})$. Let $U : \mathcal{H} \to \mathcal{H}$ be a unitary operator and G_1 :

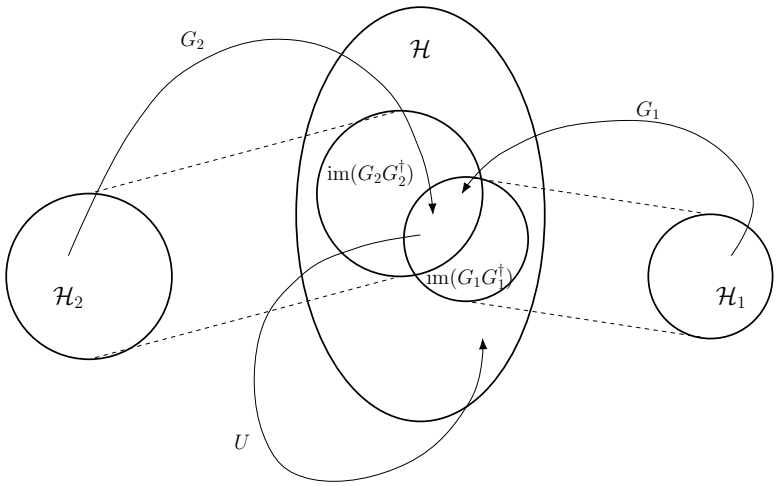

Figure 8.1: Illustration of the standard-form encoding.

$\mathcal{H}_1 \to \mathcal{H}$, $G_2 : \mathcal{H}_2 \to \mathcal{H}$ be two isometries, i.e., we have

$$G_1^\dagger G_1 = I_{\mathcal{H}_1}, \qquad G_1 G_1^\dagger = P_{\mathcal{H}_1}, \tag{8.1}$$

$$G_2^\dagger G_2 = I_{\mathcal{H}_2}, \qquad G_2 G_2^\dagger = P_{\mathcal{H}_2}, \tag{8.2}$$

$$U^\dagger U = U U^\dagger = I_{\mathcal{H}}, \tag{8.3}$$

where $P_{\mathcal{H}_j}$ is an orthogonal projection on $\mathcal{H}_j$ with image $\mathrm{im}(P_{\mathcal{H}_j}) = \mathrm{im}(G_j G_j^\dagger) = \mathrm{im}(G_j)$. We say that the isometries G_1, G_2 and the unitary U encode $G_2^\dagger U G_1$ in standard form [72]. We illustrate this abstract setting in Figure 8.1.

The notion of standard-form encoding is inspired by existing quantum simulation algorithms. Recall from Section 2.5 that the Taylor-series algorithm implements the linear combination $\sum_{\xi=0}^{\Xi-1} \beta_\xi \tilde{V}_\xi$, where $\tilde{V}_\xi$ are products of unitaries and

$\beta_\xi > 0$. This can be reformulated as a standard-form encoding by setting

$$G_1 = G_2 = \frac{1}{\sqrt{\|\beta\|_1}} \sum_{\xi=0}^{\Xi-1} \sqrt{\beta_\xi} |\xi\rangle \otimes I, \qquad U = \sum_{\xi=0}^{\Xi-1} |\xi\rangle \langle\xi| \otimes \tilde{V}_\xi, \tag{8.4}$$

which encodes the desired linear combination up to a scaled-down factor

$$G_2^\dagger U G_1 = \frac{\sum_{\xi=0}^{\Xi-1} \beta_\xi \tilde{V}_\xi}{\|\beta\|_1}. \tag{8.5}$$

Similarly, the simulation algorithm introduced in Section 2.6 considers Hamiltonians of the form $H = \sum_{\gamma=1}^{\Gamma} \alpha_\gamma H_\gamma$, which can also be recast using standard-form encoding:

$$G_1 = G_2 = \frac{1}{\sqrt{\|\alpha\|_1}} \sum_{\gamma=1}^{\Gamma} \sqrt{\alpha_\gamma} |\gamma\rangle \otimes I, \qquad U = \sum_{\gamma=1}^{\Gamma} |\gamma\rangle \langle\gamma| \otimes H_\gamma. \tag{8.6}$$

However, standard-form encoding extends these quantum simulation algorithms in that it allows arbitrary definitions of the isometries G_1, G_2 and the unitary U.

Let the singular value decomposition of $G_2^\dagger U G_1$ be

$$G_2^\dagger U G_1 = \sum_j \sigma_j |\varphi_j\rangle\langle\psi_j|, \tag{8.7}$$

where $\{|\psi_j\rangle\}_j$ and $\{|\varphi_j\rangle\}_j$ are orthonormal on $\mathcal{H}_1$ and $\mathcal{H}_2$, respectively. Further-

more, the singular values σ_j satisfy $0 \leq \sigma_j \leq 1$, since

$$\left\| G_2^\dagger U G_1 \right\| \leq \left\| G_2^\dagger \right\| \|U\| \|G_1\| = 1. \tag{8.8}$$

Then, the idea of quantum singular value transformation is as follows. We are given a standard-form encoding $A = G_2^\dagger U G_1$ as input. Our goal is to obtain an output operator B whose singular values are related to those of A by certain polynomial functions. In this sense, the singular values of A are transformed by polynomial functions to output B.

Quantum singular value transformation achieves this with a quantum circuit V on space $\mathcal{H}$, constructed from the specified polynomial functions, such that the output $B = G_2^\dagger V G_1$ is encoded in standard form. Examples of this include implementing Chebyshev polynomials for the Taylor-series algorithm [14, 15] and polynomial approximations of exponentials of trigonometric functions for the QSP algorithm [73, 74]. However, we identify applications of quantum singular value transformation to designing other quantum algorithms beyond quantum simulation.

8.2 Qubitization

Since the main goal of quantum singular value transformation is to manipulate the singular values of $G_2^\dagger U G_1$ using a quantum circuit, it is natural to ask

how $G_2^\dagger U G_1$ are related to those operations that we can perform on a quantum computer, such as $2G_1G_1^\dagger - I$, $2G_2G_2^\dagger - I$, and U. This relation concerns the spectra of operators and is made clear through qubitization, which builds on earlier results such as Szegedy quantum walk [101] and Marriott-Watrous QMA amplification [79].

Let $|\psi_j\rangle$ be the unit right singular vector of $G_2^\dagger U G_1$ with singular value σ_j and let $|\varphi_j\rangle$ be the corresponding left singular vector. We consider the following pair of subspaces

$$\mathcal{H}_{1,j} = \text{span}\left\{G_1|\psi_j\rangle, U^\dagger G_2|\varphi_j\rangle\right\}, \quad \mathcal{H}_{2,j} = \text{span}\left\{G_2|\varphi_j\rangle, UG_1|\psi_j\rangle\right\}. \tag{8.9}$$

As the following lemmas show, these subspaces are either one-dimensional or two-dimensional depending on the value of σ_j.

Lemma 47 (1D subspace pair)**.** *Let $\mathcal{H}_1$, $\mathcal{H}_2$ and $\mathcal{H}$ be vector spaces with dimensionality $\dim(\mathcal{H}_1), \dim(\mathcal{H}_2) \leq \dim(\mathcal{H})$. Let $U : \mathcal{H} \to \mathcal{H}$ be a unitary operator and $G_1 : \mathcal{H}_1 \to \mathcal{H}$, $G_2 : \mathcal{H}_2 \to \mathcal{H}$ be two isometries. Let $|\psi_j\rangle$ be the unit right singular vector of $G_2^\dagger U G_1$ with singular value σ_j and let $|\varphi_j\rangle$ be the corresponding left singular vector. Define $\mathcal{H}_{1,j} = \text{span}\left\{G_1|\psi_j\rangle, U^\dagger G_2|\varphi_j\rangle\right\}$ and $\mathcal{H}_{2,j} = \text{span}\left\{G_2|\varphi_j\rangle, UG_1|\psi_j\rangle\right\}$. The following conditions are equivalent:*

1. $\dim\left(\mathcal{H}_{1,j}\right) = 1$.

2. $\dim\left(\mathcal{H}_{2,j}\right) = 1$.

3. $\mathcal{B}_{1,j} = \{G_1|\psi_j\rangle\}$ *is a basis for* $\mathcal{H}_{1,j}$.

4. $\mathcal{B}_{2,j} = \{G_2|\varphi_j\rangle\}$ *is a basis for* $\mathcal{H}_{2,j}$.

5. $\sigma_j = 1$.

If any one of the above condition is satisfied, we have

$$\left[U\right]_{\mathcal{B}_{1,j},\mathcal{B}_{2,j}} = \left[G_1 G_1^\dagger\right]_{\mathcal{B}_{1,j}} = \left[G_2 G_2^\dagger\right]_{\mathcal{B}_{2,j}} = 1. \tag{8.10}$$

Proof. Note that U is an invertible mapping from $\mathcal{H}_{1,j} = \text{span}\{G_1|\psi_j\rangle, U^\dagger G_2|\varphi_j\rangle\}$ to $\mathcal{H}_{2,j} = \text{span}\{G_2|\varphi_j\rangle, UG_1|\psi_j\rangle\}$ with inverse $U^\dagger$, which implies that $\mathcal{H}_{1,j}$ is isomorphic to $\mathcal{H}_{2,j}$ as linear subspaces. In particular, it must be that they agree on the dimensionality. Therefore, Conditions 1 and 2 are equivalent.

The vector $G_1|\psi_j\rangle$ has unit length in $\mathcal{H}_{1,j} = \text{span}\{G_1|\psi_j\rangle, U^\dagger G_2|\varphi_j\rangle\}$, and must therefore be a basis of $\mathcal{H}_{1,j}$ if Condition 1 holds. This shows that 1 implies 3. The fact that Condition 2 implies 4 can be argued in a similar way.

Assume that Condition 3 is satisfied, i.e.,

$$U^\dagger G_2|\varphi_j\rangle = \alpha G_1|\psi_j\rangle \tag{8.11}$$

for some complex number α. We apply $G_1^\dagger$ to both sides of the above equation and get

$$\sigma_j|\psi_j\rangle = G_1^\dagger U^\dagger G_2|\varphi_j\rangle = \alpha G_1^\dagger G_1|\psi_j\rangle = \alpha|\psi_j\rangle. \tag{8.12}$$

Taking the Euclidean norm of both sides gives

$$\sigma_j = \|\sigma_j|\psi_j\rangle\| = \|\alpha|\psi_j\rangle\| = \|\alpha G_1|\psi_j\rangle\| = \left\|U^\dagger G_2|\varphi_j\rangle\right\| = 1, \tag{8.13}$$

which proves Condition 5. Similarly, Condition 4 implies 5 as well.

Finally, if $\sigma_j = 1$, we can compute

$$\begin{aligned} 1 &= \|UG_1|\psi_j\rangle\|^2 \\ &= \left\|G_2G_2^\dagger UG_1|\psi_j\rangle\right\|^2 + \left\|(I - G_2G_2^\dagger)UG_1|\psi_j\rangle\right\|^2 \\ &= \|\sigma_j G_2|\varphi_j\rangle\|^2 + \left\|(I - G_2G_2^\dagger)UG_1|\psi_j\rangle\right\|^2 \\ &= 1 + \left\|(I - G_2G_2^\dagger)UG_1|\psi_j\rangle\right\|^2, \end{aligned} \tag{8.14}$$

where the second equality follows by the Pythagorean theorem. This implies that

$$(I - G_2G_2^\dagger)UG_1|\psi_j\rangle = 0, \tag{8.15}$$

or equivalently

$$UG_1|\psi_j\rangle = G_2G_2^\dagger UG_1|\psi_j\rangle = \sigma_j G_2|\varphi_j\rangle = G_2|\varphi_j\rangle. \tag{8.16}$$

We conclude that $\dim\left(\mathcal{H}_{2,j}\right) = 1$, which implies Condition 2. The matrix representations of U, $G_1G_1^\dagger$ and $G_2G_2^\dagger$ now follow from a direct calculation. □

Lemma 48 (2D subspace pair)**.** *Let $\mathcal{H}_1$, $\mathcal{H}_2$ and $\mathcal{H}$ be vector spaces with di-*

mensionality $\dim(\mathcal{H}_1), \dim(\mathcal{H}_2) \leq \dim(\mathcal{H})$. *Let* $U : \mathcal{H} \to \mathcal{H}$ *be a unitary operator and* $G_1 : \mathcal{H}_1 \to \mathcal{H}$, $G_2 : \mathcal{H}_2 \to \mathcal{H}$ *be two isometries. Let* $|\psi_k\rangle$ *be the unit right singular vector of* $G_2^\dagger U G_1$ *with singular value* σ_k *and let* $|\varphi_k\rangle$ *be the corresponding left singular vector. Define* $\mathcal{H}_{1,k} = \operatorname{span}\left\{G_1|\psi_k\rangle, U^\dagger G_2|\varphi_k\rangle\right\}$ *and* $\mathcal{H}_{2,k} = \operatorname{span}\left\{G_2|\varphi_k\rangle, UG_1|\psi_k\rangle\right\}$. *The following conditions are equivalent:*

1. $\dim\left(\mathcal{H}_{1,k}\right) = 2$.
2. $\dim\left(\mathcal{H}_{2,k}\right) = 2$.
3. $\mathcal{B}_{1,k} = \left\{G_1|\psi_k\rangle, U^\dagger G_2|\varphi_k\rangle\right\}$ *is a basis for* $\mathcal{H}_{1,k}$.
4. $\mathcal{B}_{2,k} = \left\{G_2|\varphi_k\rangle, UG_1|\psi_k\rangle\right\}$ *is a basis for* $\mathcal{H}_{2,k}$.
5. $0 \leq \sigma_k < 1$.

If any one of the above condition is satisfied, we have

1. *With respect to the bases* $\mathcal{B}_{1,k}$ *and* $\mathcal{B}_{2,k}$,

$$\left[U\right]_{\mathcal{B}_{1,k},\mathcal{B}_{2,k}} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \left[G_1 G_1^\dagger\right]_{\mathcal{B}_{1,k}} = \begin{bmatrix} 1 & \sigma_k \\ 0 & 0 \end{bmatrix}, \quad \left[G_2 G_2^\dagger\right]_{\mathcal{B}_{2,k}} = \begin{bmatrix} 1 & \sigma_k \\ 0 & 0 \end{bmatrix}. \tag{8.17}$$

2. $\mathcal{B}'_{1,k} = \left\{G_1|\psi_k\rangle, \frac{U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}}\right\}$ *is an orthonormal basis for the subspace* $\mathcal{H}_{1,k}$ *and* $\mathcal{B}'_{2,k} = \left\{G_2|\varphi_k\rangle, \frac{UG_1|\psi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}}\right\}$ *is an orthonormal basis for* $\mathcal{H}_{2,k}$,

with respect to which

$$\left[U\right]_{\mathcal{B}'_{1,k},\mathcal{B}'_{2,k}} = \begin{bmatrix} \sigma_k & \sqrt{1-\sigma_k^2} \\ \sqrt{1-\sigma_k^2} & -\sigma_k \end{bmatrix}, \quad \left[G_1G_1^\dagger\right]_{\mathcal{B}'_{1,k}} = \left[G_2G_2^\dagger\right]_{\mathcal{B}'_{2,k}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{8.18}$$

Proof. The equivalence of the five conditions is proved in a similar way as in Lemma 47. Assuming that any one (thus all) of the condition holds, we can compute

$$\begin{aligned} G_1G_1^\dagger G_1|\psi_k\rangle &= G_1|\psi_k\rangle, \qquad G_1G_1^\dagger U^\dagger G_2|\varphi_k\rangle = \sigma_k G_1|\psi_k\rangle, \\ G_2G_2^\dagger G_2|\varphi_k\rangle &= G_2|\varphi_k\rangle, \qquad G_2G_2^\dagger U G_1|\psi_k\rangle = \sigma_k G_2|\varphi_k\rangle, \end{aligned} \tag{8.19}$$

which gives the matrix representation of $G_1G_1^\dagger$ and $G_2G_2^\dagger$. $\left[U\right]_{\mathcal{B}_{1,k},\mathcal{B}_{2,k}}$ is obtained through a direct calculation.

We now apply the Gram-Schmidt process to $\mathcal{B}_{1,k} = \left\{G_1|\psi_k\rangle, U^\dagger G_2|\varphi_k\rangle\right\}$ to construct a unit vector orthogonal to $G_1|\psi_k\rangle$

$$\begin{aligned} \frac{U^\dagger G_2|\varphi_k\rangle - G_1|\psi_k\rangle\langle\psi_k|G_1^\dagger U^\dagger G_2|\varphi_k\rangle}{\left\|U^\dagger G_2|\varphi_k\rangle - G_1|\psi_k\rangle\langle\psi_k|G_1^\dagger U^\dagger G_2|\varphi_k\rangle\right\|} &= \frac{U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\|U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle\|} \\ &= \frac{U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}}. \end{aligned} \tag{8.20}$$

The last equality holds since

$$\begin{aligned}\left\|U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle\right\|^2 &= \langle\varphi_k|G_2^\dagger UU^\dagger G_2|\varphi_k\rangle - \sigma_k\langle\varphi_k|G_2^\dagger UG_1|\psi_k\rangle \\ &\quad - \sigma_k\langle\psi_k|G_1^\dagger U^\dagger G_2|\varphi_k\rangle + \sigma_k^2\langle\psi_k|G_1^\dagger G_1|\psi_k\rangle \\ &= 1 - 2\sigma_k^2 + \sigma_k^2 = 1 - \sigma_k^2.\end{aligned} \tag{8.21}$$

In a similar way, we produce the following unit vector orthogonal to $G_2|\varphi_k\rangle$

$$\frac{UG_1|\psi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}}. \tag{8.22}$$

The matrix of U with respect to the orthonormal basis $\mathcal{B}'_{1,k} = \big\{G_1|\psi_k\rangle$, $\frac{U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}}\big\}$ and $\mathcal{B}'_{2,k} = \big\{G_2|\varphi_k\rangle, \frac{UG_1|\psi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}}\big\}$ can be computed by

$$\begin{aligned}UG_1|\psi_k\rangle &= \lambda G_2|\varphi_k\rangle + \sqrt{1-\sigma_k^2}\frac{UG_1|\psi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}}, \\ U\frac{U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}} &= \frac{G_2|\varphi_k\rangle - \sigma_k UG_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}} \\ &= \sqrt{1-\sigma_k^2}G_2|\varphi_k\rangle - \sigma_k\frac{UG_1|\psi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}},\end{aligned} \tag{8.23}$$

whereas $\big[G_1G_1^\dagger\big]_{\mathcal{B}'_{1,k}}$ and $\big[G_2G_2^\dagger\big]_{\mathcal{B}'_{2,k}}$ are obtained from

$$\begin{aligned}G_1G_1^\dagger\frac{U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}} &= \frac{\sigma_k G_1|\psi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}} = 0, \\ G_2G_2^\dagger\frac{UG_1|\psi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}} &= \frac{\sigma_k G_2|\varphi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}} = 0.\end{aligned} \tag{8.24}$$

□

We now decompose the entire space $\mathcal{H}$ into one- and two-dimensional subspaces as defined above. To this end, we consider the singular value decomposition of $G_2^\dagger U G_1$, dropping terms with zero singular values and grouping the remaining ones as

$$G_2^\dagger U G_1 = \sum_{0<\sigma_j\leq 1} \sigma_j |\varphi_j\rangle\langle\psi_j| = \sum_{\sigma_j=1} |\varphi_j\rangle\langle\psi_j| + \sum_{0<\sigma_k<1} \sigma_k |\varphi_k\rangle\langle\psi_k|. \tag{8.25}$$

This gives pairs of one-dimensional subspaces $\mathcal{H}_{1,j} = \text{span}\left\{G_1|\psi_j\rangle, U^\dagger G_2|\varphi_j\rangle\right\}$, $\mathcal{H}_{2,j} = \text{span}\left\{G_2|\varphi_j\rangle, UG_1|\psi_j\rangle\right\}$ with $\sigma_j = 1$ and pairs of two-dimensional subspaces $\mathcal{H}_{1,k} = \text{span}\left\{G_1|\psi_k\rangle, U^\dagger G_2|\varphi_k\rangle\right\}$, $\mathcal{H}_{2,k} = \text{span}\left\{G_2|\varphi_k\rangle, UG_1|\psi_k\rangle\right\}$ with $0 < \sigma_k < 1$.

In general, the orthonormal set of right singular vectors $\{|\psi_j\rangle\}_j \cup \{|\psi_k\rangle\}_k$ does not span the entire $\mathcal{H}_1$. We expand it with $\{|\bar{\psi}_l\rangle\}_l$ so that $\{|\psi_j\rangle\}_j \cup \{|\psi_k\rangle\}_k \cup \{|\bar{\psi}_l\rangle\}_l$ becomes an orthonormal basis for $\mathcal{H}_1$. Similarly, we expand $\{|\varphi_j\rangle\}_j \cup \{|\varphi_k\rangle\}_k$ with $\{|\bar{\varphi}_m\rangle\}_m$ to form an orthonormal basis for $\mathcal{H}_2$. With a slight abuse of notation, we denote

$$\begin{aligned} \bar{\mathcal{H}}_{1,l} &= \text{span}\left\{G_1|\bar{\psi}_l\rangle\right\}, \qquad \bar{\mathcal{H}}_{1,m} = \text{span}\left\{U^\dagger G_2|\bar{\varphi}_m\rangle\right\}, \\ \bar{\mathcal{H}}_{2,l} &= \text{span}\left\{UG_1|\bar{\psi}_l\rangle\right\}, \qquad \bar{\mathcal{H}}_{2,m} = \text{span}\left\{G_2|\bar{\varphi}_m\rangle\right\}. \end{aligned} \tag{8.26}$$

The following theorem claims that the entire space $\mathcal{H}$ can be decomposed as an orthogonal direct sum of the above spaces.

Theorem 49 (Qubitization by pairing subspaces)**.** *Let $\mathcal{H}_1$, $\mathcal{H}_2$ and $\mathcal{H}$ be vector spaces with dimensionality $\dim(\mathcal{H}_1), \dim(\mathcal{H}_2) \leq \dim(\mathcal{H})$. Let $U : \mathcal{H} \to \mathcal{H}$ be a unitary operator and $G_1 : \mathcal{H}_1 \to \mathcal{H}$, $G_2 : \mathcal{H}_2 \to \mathcal{H}$ be two isometries. Let $G_2^\dagger U G_1 = \sum_j \sigma_j |\varphi_j\rangle\langle\psi_j|$ with $0 < \sigma_j \leq 1$ be the singular value decomposition. Define $\mathcal{H}_{1,j}$, $\mathcal{H}_{2,j}$ with $\sigma_j = 1$, $\mathcal{H}_{1,k}$, $\mathcal{H}_{2,k}$ with $0 < \sigma_k < 1$, and $\bar{\mathcal{H}}_{1,l}$, $\bar{\mathcal{H}}_{2,l}$, $\bar{\mathcal{H}}_{1,m}$, $\bar{\mathcal{H}}_{2,m}$ as above. Then $\mathcal{H}$ admits the following pair of decompositions*

$$\begin{aligned} \mathcal{H} &= \bigodot_{\sigma_j=1} \mathcal{H}_{1,j} \bigodot_{0<\sigma_k<1} \mathcal{H}_{1,k} \bigodot_{l} \bar{\mathcal{H}}_{1,l} \bigodot_{m} \bar{\mathcal{H}}_{1,m} \bigodot \mathcal{H}_{1,\perp} \\ &= \bigodot_{\sigma_j=1} \mathcal{H}_{2,j} \bigodot_{0<\sigma_k<1} \mathcal{H}_{2,k} \bigodot_{l} \bar{\mathcal{H}}_{2,l} \bigodot_{m} \bar{\mathcal{H}}_{2,m} \bigodot \mathcal{H}_{2,\perp}, \end{aligned} \tag{8.27}$$

where $\bigodot$ denotes the orthogonal direct sum and $\mathcal{H}_{1,\perp}$/$\mathcal{H}_{2,\perp}$ is a space orthogonal to the remaining spaces of the same line. Here, subspaces in the first/second line are $G_1 G_1^\dagger$-/$G_2 G_2^\dagger$-invariant and U is an isomorphism between the corresponding subspaces with the same subscript. Furthermore,

1. *For $\sigma_j = 1$, $\mathcal{B}_{1,j} = \big\{G_1|\psi_j\rangle\big\}$/$\mathcal{B}_{2,j} = \big\{G_2|\varphi_j\rangle\big\}$ is a basis for $\mathcal{H}_{1,j}$/$\mathcal{H}_{2,j}$ and*

$$\big[U\big]_{\mathcal{B}_{1,j},\mathcal{B}_{2,j}} = \big[G_1 G_1^\dagger\big]_{\mathcal{B}_{1,j}} = \big[G_2 G_2^\dagger\big]_{\mathcal{B}_{2,j}} = 1. \tag{8.28}$$

2. *For $0 < \sigma_k < 1$, $\mathcal{B}'_{1,k} = \big\{G_1|\psi_k\rangle, \frac{U^\dagger G_2|\varphi_k\rangle - \sigma_k G_1|\psi_k\rangle}{\sqrt{1-\sigma_k^2}}\big\}$/$\mathcal{B}'_{2,k} = \big\{G_2|\varphi_k\rangle$, $\frac{U G_1|\psi_k\rangle - \sigma_k G_2|\varphi_k\rangle}{\sqrt{1-\sigma_k^2}}\big\}$ is an orthonormal basis for $\mathcal{H}_{1,k}$/$\mathcal{H}_{2,k}$, with respect to which*

$$\big[U\big]_{\mathcal{B}'_{1,k},\mathcal{B}'_{2,k}} = \begin{bmatrix} \sigma_k & \sqrt{1-\sigma_k^2} \\ \sqrt{1-\sigma_k^2} & -\sigma_k \end{bmatrix}, \quad \big[G_1 G_1^\dagger\big]_{\mathcal{B}'_{1,k}} = \big[G_2 G_2^\dagger\big]_{\mathcal{B}'_{2,k}} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{8.29}$$

3. *$\bar{\mathcal{B}}_{1,l} = \{G_1|\bar{\psi}_l\rangle\}/\bar{\mathcal{B}}_{2,l} = \{UG_1|\bar{\psi}_l\rangle\}$ is a basis for $\bar{\mathcal{H}}_{1,l}/\bar{\mathcal{H}}_{2,l}$, and $\bar{\mathcal{B}}_{1,m} = \{U^\dagger G_2|\bar{\varphi}_m\rangle\}/\bar{\mathcal{B}}_{2,m} = \{G_2|\bar{\varphi}_m\rangle\}$ is a basis for $\bar{\mathcal{H}}_{1,m}/\bar{\mathcal{H}}_{2,m}$, under which*

$$\begin{aligned} \big[U\big]_{\bar{\mathcal{B}}_{1,l},\bar{\mathcal{B}}_{2,l}} &= \big[U\big]_{\bar{\mathcal{B}}_{1,m},\bar{\mathcal{B}}_{2,m}} = [1], \\ \big[G_1 G_1^\dagger\big]_{\bar{\mathcal{B}}_{1,l}} &= \big[G_2 G_2^\dagger\big]_{\bar{\mathcal{B}}_{2,m}} = [1], \\ \big[G_1 G_1^\dagger\big]_{\bar{\mathcal{B}}_{1,m}} &= \big[G_2 G_2^\dagger\big]_{\bar{\mathcal{B}}_{2,l}} = [0]. \end{aligned} \tag{8.30}$$

4. *Restricted $\mathcal{H}_{1,\perp}$ and $\mathcal{H}_{2,\perp}$, U is an isometry; $G_1 G_1^\dagger$ and $G_2 G_2^\dagger$ are zero.*

Proof. Let $G_2^\dagger U G_1 = \sum_j \sigma_j |\varphi_j\rangle\langle\psi_j|$ with $0 < \sigma_j \leq 1$ be the singular value decomposition where terms with singular value zero are dropped. Define $\mathcal{H}_{1,j}$, $\mathcal{H}_{2,j}$, $\mathcal{H}_{1,k}$, $\mathcal{H}_{2,k}$, $\bar{\mathcal{H}}_{1,l}$, $\bar{\mathcal{H}}_{2,l}$, $\bar{\mathcal{H}}_{1,m}$, and $\bar{\mathcal{H}}_{2,m}$ as stated. We now prove the orthogonality claim in (8.27):

- $\mathcal{H}_{1,j} \perp \mathcal{H}_{1,j'}$ and $\mathcal{H}_{2,j} \perp \mathcal{H}_{2,j'}$ for $j \neq j'$:

$$\langle\psi_{j'}|G_1^\dagger G_1|\psi_j\rangle = \langle\psi_{j'}|\psi_j\rangle = 0, \tag{8.31}$$

and

$$\langle\varphi_{j'}|G_2^\dagger G_2|\varphi_j\rangle = \langle\varphi_{j'}|\varphi_j\rangle = 0; \tag{8.32}$$

- $\mathcal{H}_{1,k}\perp\mathcal{H}_{1,k'}$ and $\mathcal{H}_{2,k}\perp\mathcal{H}_{2,k'}$ for $k \neq k'$:

$$\begin{aligned}
\langle\psi_{k'}|G_1^\dagger G_1|\psi_k\rangle &= \langle\psi_{k'}|\psi_k\rangle = 0,\\
\langle\psi_{k'}|G_1^\dagger U^\dagger G_2|\varphi_k\rangle &= \sigma_k\langle\psi_{k'}|\psi_k\rangle = 0,\\
\langle\varphi_{k'}|G_2^\dagger U G_1|\psi_k\rangle &= \sigma_k\langle\varphi_{k'}|\varphi_k\rangle = 0,\\
\langle\varphi_{k'}|G_2^\dagger U U^\dagger G_2|\varphi_k\rangle &= \langle\varphi_{k'}|\varphi_k\rangle = 0,
\end{aligned} \tag{8.33}$$

and

$$\begin{aligned}
\langle\varphi_{k'}|G_2^\dagger G_2|\varphi_k\rangle &= \langle\varphi_{k'}|\varphi_k\rangle = 0,\\
\langle\varphi_{k'}|G_2^\dagger U G_1|\psi_k\rangle &= \sigma_k\langle\varphi_{k'}|\varphi_k\rangle = 0,\\
\langle\psi_{k'}|G_1^\dagger U^\dagger G_2|\varphi_k\rangle &= \sigma_k\langle\psi_{k'}|\psi_k\rangle = 0,\\
\langle\psi_{k'}|G_1^\dagger U^\dagger U G_1|\psi_k\rangle &= \langle\psi_{k'}|\psi_k\rangle = 0;
\end{aligned} \tag{8.34}$$

- $\bar{\mathcal{H}}_{1,l}\perp\bar{\mathcal{H}}_{1,l'}$ and $\bar{\mathcal{H}}_{2,l}\perp\bar{\mathcal{H}}_{2,l'}$ for $l \neq l'$:

$$\langle\bar{\psi}_{l'}|G_1^\dagger G_1|\bar{\psi}_l\rangle = \langle\bar{\psi}_{l'}|\bar{\psi}_l\rangle = 0, \tag{8.35}$$

and

$$\langle\bar{\psi}_{l'}|G_1^\dagger U^\dagger U G_1|\bar{\psi}_l\rangle = \langle\bar{\psi}_{l'}|\bar{\psi}_l\rangle = 0; \tag{8.36}$$

- $\bar{\mathcal{H}}_{1,m} \perp \bar{\mathcal{H}}_{1,m'}$ and $\bar{\mathcal{H}}_{2,m} \perp \bar{\mathcal{H}}_{2,m'}$ for $m \neq m'$:

$$\langle \bar{\varphi}_{m'} | G_2^\dagger U U^\dagger G_2 | \bar{\varphi}_m \rangle = \langle \bar{\varphi}_{m'} | \bar{\varphi}_m \rangle = 0, \tag{8.37}$$

and

$$\langle \bar{\varphi}_{m'} | G_2^\dagger G_2 | \bar{\varphi}_m \rangle = \langle \bar{\varphi}_{m'} | \bar{\varphi}_m \rangle = 0; \tag{8.38}$$

- $\mathcal{H}_{1,j} \perp \mathcal{H}_{1,k}$ and $\mathcal{H}_{2,j} \perp \mathcal{H}_{2,k}$:

$$\begin{aligned} \langle \psi_j | G_1^\dagger G_1 | \psi_k \rangle &= \langle \psi_j | \psi_k \rangle = 0, \\ \langle \psi_j | G_1^\dagger U^\dagger G_2 | \varphi_k \rangle &= \sigma_k \langle \psi_j | \psi_k \rangle = 0, \end{aligned} \tag{8.39}$$

and

$$\begin{aligned} \langle \varphi_j | G_2^\dagger G_2 | \varphi_k \rangle &= \langle \varphi_j | \varphi_k \rangle = 0, \\ \langle \varphi_j | G_2^\dagger U G_1 | \psi_k \rangle &= \sigma_k \langle \varphi_j | \varphi_k \rangle = 0; \end{aligned} \tag{8.40}$$

- $\mathcal{H}_{1,j} \perp \bar{\mathcal{H}}_{1,l}$ and $\mathcal{H}_{2,j} \perp \bar{\mathcal{H}}_{2,l}$:

$$\langle \psi_j | G_1^\dagger G_1 | \bar{\psi}_l \rangle = \langle \psi_j | \bar{\psi}_l \rangle = 0, \tag{8.41}$$

and

$$\langle \varphi_j | G_2^\dagger U G_1 | \bar{\psi}_l \rangle = \sigma_j \langle \psi_j | \bar{\psi}_l \rangle = 0; \tag{8.42}$$

- $\mathcal{H}_{1,j} \perp \bar{\mathcal{H}}_{1,m}$ and $\mathcal{H}_{2,j} \perp \bar{\mathcal{H}}_{2,m}$:

$$\langle \psi_j | G_1^\dagger U^\dagger G_2 | \bar{\varphi}_m \rangle = \sigma_j \langle \varphi_j | \bar{\varphi}_m \rangle = 0, \tag{8.43}$$

and

$$\langle \varphi_j | G_2^\dagger G_2 | \bar{\varphi}_m \rangle = \langle \varphi_j | \bar{\varphi}_m \rangle = 0; \tag{8.44}$$

- $\mathcal{H}_{1,k} \perp \bar{\mathcal{H}}_{1,l}$ and $\mathcal{H}_{2,k} \perp \bar{\mathcal{H}}_{2,l}$:

$$\begin{aligned} \langle \psi_k | G_1^\dagger G_1 | \bar{\psi}_l \rangle &= \langle \psi_k | \bar{\psi}_l \rangle = 0, \\ \langle \varphi_k | G_2^\dagger U G_1 | \bar{\psi}_l \rangle &= \sigma_k \langle \psi_k | \bar{\psi}_l \rangle = 0, \end{aligned} \tag{8.45}$$

and

$$\begin{aligned} \langle \varphi_k | G_2^\dagger U G_1 | \bar{\psi}_l \rangle &= \sigma_k \langle \psi_k | \bar{\psi}_l \rangle = 0, \\ \langle \psi_k | G_1^\dagger U^\dagger U G_1 | \bar{\psi}_l \rangle &= \langle \psi_k | \bar{\psi}_l \rangle = 0; \end{aligned} \tag{8.46}$$

- $\mathcal{H}_{1,k} \perp \bar{\mathcal{H}}_{1,m}$ and $\mathcal{H}_{2,k} \perp \bar{\mathcal{H}}_{2,m}$:

$$\begin{aligned} \langle \psi_k | G_1^\dagger U^\dagger G_2 | \bar{\varphi}_m \rangle &= \sigma_k \langle \varphi_k | \bar{\varphi}_m \rangle = 0, \\ \langle \varphi_k | G_2^\dagger U U^\dagger G_2 | \bar{\varphi}_m \rangle &= \langle \varphi_k | \bar{\varphi}_m \rangle = 0, \end{aligned} \tag{8.47}$$

and

$$\begin{aligned} \langle \varphi_k | G_2^\dagger G_2 | \bar{\varphi}_m \rangle &= \langle \varphi_k | \bar{\varphi}_m \rangle = 0, \\ \langle \psi_k | G_1^\dagger U^\dagger G_2 | \bar{\varphi}_m \rangle &= \sigma_k \langle \varphi_k | \bar{\varphi}_m \rangle = 0; \end{aligned} \tag{8.48}$$

- $\bar{\mathcal{H}}_{1,l} \perp \bar{\mathcal{H}}_{1,m}$ and $\bar{\mathcal{H}}_{2,l} \perp \bar{\mathcal{H}}_{2,m}$: $|\bar{\psi}_l\rangle \in \ker\left(G_2^\dagger U G_1\right)$.

The claim that subspaces in the first/second line of (8.27) are $G_1G_1^\dagger$-/$G_2G_2^\dagger$-invariant and U is an isomorphism between the corresponding subspaces with the same subscript follows from Lemma 47, Lemma 48 and direct verification. Since $G_1G_1^\dagger/G_2G_2^\dagger$ is normal, it is also invariant on the orthogonal complement $\mathcal{H}_{1,\perp}/\mathcal{H}_{2\perp}$ [91, Exercise 10.7]; for the same reason, U is an isomorphism between $\mathcal{H}_{1,\perp}$ and $\mathcal{H}_{2,\perp}$.

The claimed matrix representation of U and $G_1G_1^\dagger/G_2G_2^\dagger$ follows by Lemma 47, Lemma 48 and direct calculation. For the orthogonal complement $\mathcal{H}_{1,\perp}$, we have

$$\begin{aligned}
\mathcal{H}_{1,\perp} &= \left[\bigodot_{\sigma_j=1} \mathcal{H}_{1,j} \bigodot_{0<\sigma_k<1} \mathcal{H}_{1,k} \bigodot_{l} \bar{\mathcal{H}}_{1,l} \bigodot_{m} \bar{\mathcal{H}}_{1,m}\right]^\perp \\
&\subseteq \left[\bigodot_{\sigma_j=1} \operatorname{span}\left\{G_1|\psi_j\rangle\right\} \bigodot_{0<\sigma_k<1} \operatorname{span}\left\{G_1|\psi_k\rangle\right\} \bigodot_{l} \operatorname{span}\left\{G_1|\bar{\psi}_l\rangle\right\}\right]^\perp \\
&= \left[G_1 \operatorname{span}\left\{\left\{|\psi_j\rangle\right\}_j \cup \left\{|\psi_k\rangle\right\}_k \cup \left\{|\bar{\psi}_l\rangle\right\}_l\right\}\right]^\perp \\
&= \left[\operatorname{im}(G_1)\right]^\perp = \left[\operatorname{im}(G_1G_1^\dagger)\right]^\perp = \ker(G_1G_1^\dagger).
\end{aligned} \tag{8.49}$$

Hence $G_1G_1^\dagger$ is zero on $\mathcal{H}_{1,\perp}$; the claim that $G_2G_2^\dagger$ is zero on $\mathcal{H}_{2,\perp}$ is proved in a similar way. Restricted to $\mathcal{H}_{1,\perp}$ and $\mathcal{H}_{2,\perp}$, U still preserves the inner product and is therefore an isometry. □

8.3 Quantum signal processing

The final component of quantum singular value transformation is quantum signal processing, achieved by alternating between unitary $U/U^\dagger$ and partial rotation $e^{i\phi(2G_1G_1^\dagger - I)}/e^{i\phi(2G_2G_2^\dagger - I)}$. Specifically, for any real vector $\vec{\phi} = [\phi_1, \ldots, \phi_m] \in \mathbb{R}^m$, we define

$$V_{\vec{\phi}} = \begin{cases} e^{i\phi_1(2G_2G_2^\dagger - I)} U \prod_{j=1}^{\frac{m-1}{2}} \left(e^{i\phi_{2j}(2G_1G_1^\dagger - I)} U^\dagger e^{i\phi_{2j+1}(2G_2G_2^\dagger - I)} U \right) & \text{for odd } m, \\ \prod_{j=1}^{\frac{m}{2}} \left(e^{i\phi_{2j-1}(2G_1G_1^\dagger - I)} U^\dagger e^{i\phi_{2j}(2G_2G_2^\dagger - I)} U \right) & \text{for even } m. \end{cases} \tag{8.50}$$

Then, for certain even/odd polynomials $f : \mathbb{R} \to \mathbb{C}$ [50, Corollary 8], we can find $[\phi_1, \ldots, \phi_m] \in \mathbb{R}^m$ so that G_1, G_2, and $V_{\vec{\phi}}$ encode an operator whose singular values are related to those of the input $G_2^\dagger U G_1 = \sum_j \sigma_j |\varphi_j\rangle\langle\psi_j|$ by f. Specifically, we have

$$G_2^\dagger V_{\vec{\phi}} G_1 = \sum_j f(\sigma_j) |\varphi_j\rangle\langle\psi_j| \tag{8.51}$$

if f is an odd polynomial and

$$G_1^\dagger V_{\vec{\phi}} G_1 = \sum_j f(\sigma_j) |\psi_j\rangle\langle\psi_j| + f(0) \sum_l |\bar{\psi}_l\rangle\langle\bar{\psi}_l| \tag{8.52}$$

if f is even.

This result can be proved in a similar way as in [74, 77] for quantum simulation. The difference is that we are now manipulating the singular values of the

input operator $G_2^\dagger U G_1$ and we need to use Theorem 49 to replace the previous [74, Lemma 8 and Corollary 9]. The above result can also be extended to implement singular value transformations with real polynomials. We do not discuss these in detail and refer the reader to [50] for a complete treatment.

Instead, we now consider circuit implementation of the quantum singular value transformation. We know from (8.50) that the circuit consists of the unitary U ($U^\dagger$) and the partial rotation $e^{i\phi(2G_1G_1^\dagger - I)}$ ($e^{i\phi(2G_2G_2^\dagger - I)}$). The operator U is given as input and assumed to be directly implementable. It then remains to implement the partial rotation $e^{i\phi(2G_1G_1^\dagger - I)}$. This can be difficult if G_1 is a general isometry. However, for many applications, the isometry G_1 is defined by a state preparation procedure as

$$G_1 = \bar{G}_1 |0\rangle . \tag{8.53}$$

We then have

$$e^{i\phi(2G_1G_1^\dagger - I)} = \bar{G}_1 e^{i\phi(2|0\rangle\langle 0| - I)} \bar{G}_1{}^\dagger = \bar{G}_1 \left(e^{i\phi} |0\rangle \langle 0| + e^{-i\phi} \sum_{y \neq 0} |y\rangle \langle y| \right) \bar{G}_1{}^\dagger, \tag{8.54}$$

where each operator can be directly implemented using a quantum circuit. Figure 8.2 provides a possible circuit implementation of the entire quantum singular value transformation.

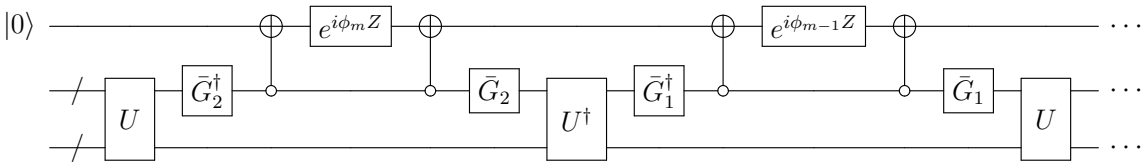

Figure 8.2: Circuit implementation of quantum singular value transformation.

8.4 Implementing principal component regression

The ability to transform singular values is central to the operation of many machine learning methods. Many quantum algorithms for basic machine learning problems, such as ordinary least squares, weighted least squares, generalized least squares, were studied in a series of works [27, 32, 55, 111]. We do not examine these problems case-by-case, but point out that they can all be reduced to implementing the Moore-Penrose pseudoinverse of the input operator, which can then be realized by performing singular value transformation with a polynomial approximation of the inverse function.

Here, we briefly discuss a new application to principal component regression. The problem of principal component regression can be formally stated as follows [48]: given a matrix $A \in \mathbb{R}^{n \times d}$, a vector $b \in \mathbb{R}^n$ and a threshold value $0 < \varsigma$, find $x \in \mathbb{R}^d$ such that

$$x = \text{argmin}_{x \in \mathbb{R}^d} \left\| \widetilde{P}_{\geq \varsigma} A P_{\geq \varsigma} x - b \right\|, \tag{8.55}$$

where $\widetilde{P}_{\geq \varsigma}, P_{\geq \varsigma}$ denote left and right singular value threshold projectors, i.e., projectors whose image is spanned by left and right singular vectors with singular

values greater than ς.

A closed-form expression for the optimal solution of (8.55) is given by $x = P_{\geq\varsigma} A^{+} \widetilde{P}_{\geq\varsigma} b = A^{+} \widetilde{P}_{\geq\varsigma} b$, where A^{+} is the Moore-Penrose pseudoinverse of A. The problem can thus be solved by singular value transformation with polynomial approximations of the inverse function [50, Theorem 41] and the threshold function [50, Theorem 31]. Suppose that operator A is given by the standard-form encoding $A = G_2^{\dagger} U G_1$ and that the spectral gap is γ: b does not overlap with left singular vectors of A with singular values in $[\varsigma - \gamma, \varsigma + \gamma]$. Then, quantum singular value transformation gives an algorithm that implements principal component regression with accuracy ϵ using $\mathcal{O}\left(\frac{1}{\gamma} \log\left(\frac{1}{\epsilon}\right)\right)$ applications of $U/U^{\dagger}$ and $e^{i\phi(2G_1 G_1^{\dagger} - I)}/e^{i\phi(2G_2 G_2^{\dagger} - I)}$. In comparison, the classical algorithm of [48] has runtime $\mathcal{O}\left(\frac{1}{\gamma^2} \log\left(\frac{1}{\epsilon}\right)\right)$.

Chapter 9: Application to Monte Carlo methods

In this chapter, we apply our analysis of product formulas (Chapter 6) to improving the performance of quantum Monte Carlo simulation. Here, the goal is to approximate certain properties of the Hamiltonian, such as the partition function, rather than simulating the full dynamics. We consider two specific systems: the transverse field Ising model (Section 9.1) and the ferromagnetic quantum spin systems (Section 9.2). For both simulations, the ideal evolution is decomposed using the second-order Suzuki formula and we show that such a decomposition can be made more efficient using our tightened analysis.

This chapter is partly based on the following paper:

[37] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu, *A theory of Trotter error*, 2019, arXiv:1912.08854.

9.1 Transverse field Ising model

Consider the following n-qubit transverse field Ising model:

$$H = -A - B, \quad A = \sum_{1 \le u < v \le n} j_{u,v} Z_u Z_v, \quad B = \sum_{1 \le u \le n} h_u X_u. \tag{9.1}$$

Here, X_u and Z_u are Pauli operators acting on the uth qubit, and $j_{u,v} \ge 0$ and $h_u \ge 0$ are nonnegative coefficients. Define $j := \max\{j_{u,v}, h_u\}$ to be the maximum

norm of the interactions. Our goal is to approximate the partition function

$$\mathcal{Z} = \mathrm{Tr}\big(e^{-H}\big) \tag{9.2}$$

up to a multiplicative error $0 < \epsilon < 1$.

Reference [20] solves this problem with an efficient classical algorithm. A key step in their algorithm is a decomposition of the evolution operator using the second-order Suzuki formula, so that

$$\mathcal{Z}' = \mathrm{Tr}\big(e^{\frac{1}{2r}A} e^{\frac{1}{r}B} e^{\frac{1}{2r}A}\big)^r \lessapprox (1+\epsilon)\mathrm{Tr}\big(e^{-H}\big) = (1+\epsilon)\mathcal{Z}. \tag{9.3}$$

However, their original analysis does not exploit the commutativity relation between A and B, and can be improved by the techniques developed here.

Note that this is different from the usual setting of digital quantum simulation. Indeed, as the matrix exponentials in the product formula are no longer unitary, we will introduce an additional multiplicative factor when we apply Theorem 39. Also, we need to estimate the multiplicative error as opposed to the additive error of the Trotter decomposition, which is addressed by the following lemma.

Lemma 50 (Relative perturbation of eigenvalues [43, Theorem 2.1] [63, Theorem 5.4])**.** *Let matrix C be positive semidefinite and D be nonsingular. Assume that*

the eigenvalues $\lambda_i(C)$ *and* $\lambda_i(D^\dagger CD)$ *are ordered non-increasingly. Then,*

$$\lambda_i(D^\dagger CD) \leq \lambda_i(C) \left\| D^\dagger D \right\|. \tag{9.4}$$

Let A and B be Hermitian matrices and consider the evolution $e^{t(A+B)}$ with $t \geq 0$. Our goal is to choose r sufficiently large so that the eigenvalues are approximated as

$$\lambda_i\left(\left(e^{\frac{t}{2r}A} e^{\frac{t}{r}B} e^{\frac{t}{2r}A}\right)^r\right) \approx \lambda_i\left(e^{t(A+B)}\right) \tag{9.5}$$

up to a small multiplicative error. We define

$$\begin{aligned} U &:= e^{\frac{t}{r}(A+B)}, \\ V &:= e^{\frac{t}{2r}A} e^{\frac{t}{r}B} e^{\frac{t}{2r}A}, \\ W &:= \exp_{\mathcal{T}}\left(\int_0^{\frac{t}{r}} \mathrm{d}\tau \; e^{-\tau(A+B)} \left[e^{\frac{\tau}{2}A} B e^{-\frac{\tau}{2}A} - B + e^{\frac{\tau}{2}A} e^{\tau B} \frac{A}{2} e^{-\tau B} e^{-\frac{\tau}{2}A} - \frac{A}{2}\right] e^{\tau(A+B)}\right). \end{aligned} \tag{9.6}$$

Then, both U and V are positive semidefinite operators and we know from Theorem 33 that $V = UW$.

To analyze the operator W, we further compute

$$e^{\frac{\tau}{2}A}Be^{-\frac{\tau}{2}A} - B = \left[\frac{A}{2}, B\right]\tau + \int_0^\tau \mathrm{d}\tau_2 \int_0^{\tau_2} \mathrm{d}\tau_3 \; e^{\frac{\tau_3}{2}A}\left[\frac{A}{2}, \left[\frac{A}{2}, B\right]\right]e^{-\frac{\tau_3}{2}A}, \tag{9.7}$$

$$\begin{aligned} e^{\frac{\tau}{2}A}e^{\tau B}\frac{A}{2}e^{-\tau B}e^{-\frac{\tau}{2}} - \frac{A}{2} &= e^{\frac{\tau}{2}A}\left(\int_0^\tau \mathrm{d}\tau_2 \; e^{\tau_2 B}\left[B, \frac{A}{2}\right]e^{-\tau_2 B}\right)e^{-\frac{\tau}{2}A} \\ &= e^{\frac{\tau}{2}A}\left(\left[B, \frac{A}{2}\right]\tau + \int_0^\tau \mathrm{d}\tau_2 \int_0^{\tau_2} \mathrm{d}\tau_3 \; e^{\tau_3 B}\left[B, \left[B, \frac{A}{2}\right]\right]e^{-\tau_3 B}\right)e^{-\frac{\tau}{2}A} \\ &= \left[B, \frac{A}{2}\right]\tau + \tau\int_0^\tau \mathrm{d}\tau_2 \; e^{\frac{\tau_2}{2}A}\left[\frac{A}{2}, \left[B, \frac{A}{2}\right]\right]e^{-\frac{\tau_2}{2}A} \\ &\quad + \int_0^\tau \mathrm{d}\tau_2 \int_0^{\tau_2} \mathrm{d}\tau_3 \; e^{\frac{\tau}{2}A}e^{\tau_3 B}\left[B, \left[B, \frac{A}{2}\right]\right]e^{-\tau_3 B}e^{-\frac{\tau}{2}A}. \end{aligned} \tag{9.8}$$

By Lemma 4, we have the following upper bound on $\|W\|$:

$$\begin{aligned} &\exp\left(e^{2\frac{t}{r}\|H\|+\frac{t}{r}\|A\|}\frac{t^3}{24r^3}\left\|[A,[A,B]]\right\| + e^{2\frac{t}{r}\|H\|+\frac{t}{r}\|A\|}\frac{t^3}{12r^3}\left\|[A,[A,B]]\right\| + e^{3\frac{t}{r}\|H\|+\frac{t}{r}\|B\|}\frac{t^3}{12r^3}\left\|[B,[B,A]]\right\|\right) \\ &\leq \exp\left(\left(\frac{t^3}{4r^3}\left\|[A,[A,B]]\right\| + \frac{t^3}{12r^3}\left\|[B,[B,A]]\right\|\right)e^{4\frac{t}{r}(\|A\|+\|B\|)}\right) \end{aligned} \tag{9.9}$$

This bound is tighter than the previous result of [20, Lemma 3] in that it exploits the commutativity of operator summands.

Our goal is to bound the eigenvalues $\lambda_i\big(V^r\big)$ in terms of $\lambda_i\big(U^r\big)$. This can be done recursively as follows. We first replace the right-most V by UW and the left-most V by $W^\dagger U$. Invoking Lemma 50, we have

$$\lambda_i\big(V^r\big) = \lambda_i\big(W^\dagger U V^{r-2} U W\big) \leq \lambda_i\big(U V^{r-2} U\big)\left\|W\right\|^2. \tag{9.10}$$

By [60, Theorem 1.3.22],

$$\lambda_i\big(UV^{r-2}U\big) = \lambda_i\big(V^{\frac{r}{2}-1}UUV^{\frac{r}{2}-1}\big). \tag{9.11}$$

We now apply a similar procedure to obtain

$$\begin{aligned}
\lambda_i\big(V^{\frac{r}{2}-1}UUV^{\frac{r}{2}-1}\big) &= \lambda_i\big(W^\dagger UV^{\frac{r}{2}-2}UUV^{\frac{r}{2}-2}UW\big) \\
&\le \lambda_i\big(UV^{\frac{r}{2}-2}UUV^{\frac{r}{2}-2}U\big)\,\|W\|^2 \\
&= \lambda_i\big(V^{\frac{r}{4}-1}UUV^{\frac{r}{2}-2}UUV^{\frac{r}{4}-1}\big)\,\|W\|^2 \\
&\le \lambda_i\big(UV^{\frac{r}{4}-2}UUV^{\frac{r}{2}-2}UUV^{\frac{r}{4}-2}U\big)\,\|W\|^4 \\
&= \lambda_i\big(V^{\frac{r}{4}-1}UUV^{\frac{r}{4}-2}UUV^{\frac{r}{4}-2}UUV^{\frac{r}{4}-1}\big)\,\|W\|^4 \\
&\le \lambda_i\big(UV^{\frac{r}{4}-2}UUV^{\frac{r}{4}-2}UUV^{\frac{r}{4}-2}UUV^{\frac{r}{4}-2}U\big)\,\|W\|^6 .
\end{aligned} \tag{9.12}$$

To ensure that this recursion is valid, we choose r to be a power of two. Since any positive integer is between 2^m and 2^{m+1} for some $m \ge 0$, this choice only enlarges r by a factor of at most two. Overall,

$$\lambda_i\big(V^r\big) \le \lambda_i\big(U^r\big)\,\|W\|^r . \tag{9.13}$$

We know that

$$\|W\|^r \le \exp\left(\left(\frac{t^3}{4r^2}\left\|\big[A,\big[A,B\big]\big]\right\| + \frac{t^3}{12r^2}\left\|\big[B,\big[B,A\big]\big]\right\|\right)e^{4\frac{t}{r}(\|A\|+\|B\|)}\right). \tag{9.14}$$

We first choose

$$r \geq 4t\big(\,\|A\| + \|B\|\,\big) \tag{9.15}$$

so that $e^{4\frac{t}{r}(\|A\|+\|B\|)} \leq e < 4$. We then set

$$r \geq \max\left\{\sqrt{\frac{2t^3}{\epsilon}\left\|\big[A,\big[A,B\big]\big]\right\|}, \sqrt{\frac{2t^3}{3\epsilon}\left\|\big[B,\big[B,A\big]\big]\right\|}\right\} \tag{9.16}$$

so that both $\frac{t^3}{4r^2}\left\|\big[A,\big[A,B\big]\big]\right\|$ and $\frac{t^3}{12r^2}\left\|\big[B,\big[B,A\big]\big]\right\|$ are bounded by $\epsilon/8$. Therefore, we have $\|W^r\| \leq e^{\epsilon}$ as long as r is a power of two satisfying

$$r \geq \max\left\{4t\big(\,\|A\| + \|B\|\,\big), \sqrt{\frac{2t^3}{\epsilon}\left\|\big[A,\big[A,B\big]\big]\right\|}, \sqrt{\frac{2t^3}{3\epsilon}\left\|\big[B,\big[B,A\big]\big]\right\|}\right\}, \tag{9.17}$$

which implies

$$\mathcal{Z}' = \sum_i \lambda_i\big(V^r\big) \leq \sum_i \lambda_i\big(U^r\big)e^{\epsilon} \approx (1+\epsilon)\sum_i \lambda_i\big(U^r\big) = (1+\epsilon)\mathcal{Z} \tag{9.18}$$

assuming $\epsilon \ll 1$. Following similar arguments, we can show that this choice of r also gives a lower bound of $\mathcal{Z}'$ with $\mathcal{Z}' \geq (1-\epsilon)\mathcal{Z}$. Therefore, we have approximated the partition function up to a multiplicative error ϵ.

We now specialize our result to the transverse field Ising Hamiltonian with $t = 1$. We find that $\|A\| = \mathcal{O}\left(n^2 j\right)$, $\|B\| = \mathcal{O}\left(nj\right)$, $\left\|\big[A,\big[A,B\big]\big]\right\| = \mathcal{O}\left(n^3 j^3\right)$, and

$\left\| \left[B, \left[B, A\right]\right] \right\| = \mathcal{O}\left(n^2 j^3\right)$, which implies

$$r = \mathcal{O}\left(n^2 j + n^{3/2} j^{3/2} \epsilon^{-1/2}\right). \tag{9.19}$$

By [20, p. 17], this gives a fully polynomial randomized approximation scheme (FPRAS) with running time

$$\widetilde{\mathcal{O}}\left(n^{17} r^{14} \epsilon^{-2}\right) = \widetilde{\mathcal{O}}\left(n^{45} j^{14} \epsilon^{-2} + n^{38} j^{21} \epsilon^{-9}\right), \tag{9.20}$$

improving over the previous complexity of

$$\widetilde{\mathcal{O}}\left(n^{59} j^{21} \epsilon^{-9}\right). \tag{9.21}$$

9.2 Quantum ferromagnets

We now apply our technique to improve the Monte Carlo simulation of ferromagnetic quantum spin systems [21]. Such systems are described by the n-qubit Hamiltonian

$$H = \sum_{1 \leq u < v \leq n} \left(-b_{uv} X_u X_v + c_{uv} Y_u Y_v\right) + \sum_{u=1}^{n} d_u \left(I + Z_u\right), \tag{9.22}$$

where $0 \leq b_{uv} \leq 1$, $-b_{uv} \leq c_{uv} \leq b_{uv}$, and $-1 \leq d_{uv} \leq 1$. It will be convenient to rewrite these Hamiltonians using the coefficients $p_{uv} = (b_{uv} - c_{uv})/2$ and $q_{uv} =$

$(b_{uv} + c_{uv})/2$ as

$$H = \sum_{1 \le u < v \le n} p_{uv}\big(-X_u X_v - Y_u Y_v\big) + \sum_{1 \le u < v \le n} q_{uv}\big(-X_u X_v + Y_u Y_v\big) + \sum_{u=1}^{n} d_u\big(I + Z_u\big). \tag{9.23}$$

Since $|c_{uv}| \le b_{uv} \le 1$, we have $p_{uv}, q_{uv} \in [0, 1]$.

Our goal is to approximate the partition function

$$\mathcal{Z}(\beta, H) = \mathrm{Tr}\big[e^{-\beta H}\big] \tag{9.24}$$

for $\beta > 0$. Following the setting of [21], we restrict ourself to the n-qubit gate set

$$\left\{ f_u\big(e^{\pm t}\big), g_{uv}(t), h_{uv}(t) \;\middle|\; u, v = 1, \ldots, n,\ u \ne v,\ 0 < t < \frac{1}{2} \right\}, \tag{9.25}$$

where

$$f\big(e^{\pm t}\big) = \begin{bmatrix} e^{\pm t} & 0 \\ 0 & 1 \end{bmatrix}, \qquad g(t) = \begin{bmatrix} 1 + t^2 & 0 & 0 & t \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t & 0 & 0 & 1 \end{bmatrix}, \qquad h(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 + t^2 & t & 0 \\ 0 & t & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9.26}$$

and the subscripts u, v indicate the qubits on which the gates act nontrivially. These gates approximate the exponentials of terms of the original Hamiltonian.

Specifically, we represent the gates as

$$f_u\big(e^{\pm t}\big) = e^{\pm \frac{t}{2} F_u}, \quad g_{uv}(t) = e^{-\frac{t}{2}\widetilde{\mathscr{G}}_{uv}(t)}, \quad h_{uv}(t) = e^{-\frac{t}{2}\widetilde{\mathscr{H}}_{uv}(t)}, \tag{9.27}$$

where $0 < t < 1/2$ and

$$F_u = (I+Z_u), \quad \widetilde{\mathscr{G}}_{uv}(t) = (-X_uX_v+Y_uY_v)-\frac{2}{t}\mathscr{G}_{uv}(t), \quad \widetilde{\mathscr{H}}_{uv}(t) = (-X_uX_v-Y_uY_v)-\frac{2}{t}\mathscr{H}_{uv}(t). \tag{9.28}$$

By [21, Proposition 1], we have $\|\mathscr{G}_{uv}(t)\| \le t^2$, and $\|\mathscr{H}_{uv}(t)\| \le t^2$.

We divide the evolution into r steps and apply the second-order Suzuki formula within each step, obtaining

$$\begin{aligned} e^{-\frac{\beta}{r}H} \approx & \prod_{1\le u\le n} e^{-\frac{\beta}{2r}d_u(I+Z_u)} \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r}q_{uv}(-X_uX_v+Y_uY_v)} \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r}p_{uv}(-X_uX_v-Y_uY_v)} \\ & \cdot \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r}p_{uv}(-X_uX_v-Y_uY_v)} \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r}q_{uv}(-X_uX_v+Y_uY_v)} \prod_{1\le u\le n} e^{-\frac{\beta}{2r}d_u(I+Z_u)} \\ \approx & \prod_{1\le u\le n} f_u\big(e^{-\frac{\beta}{r}d_u}\big) \prod_{1\le u<v\le n} g_{uv}\left(\frac{\beta}{r}q_{uv}\right) \prod_{1\le u<v\le n} h_{uv}\left(\frac{\beta}{r}p_{uv}\right) \\ & \cdot \prod_{1\le u<v\le n} h_{uv}\left(\frac{\beta}{r}p_{uv}\right) \prod_{1\le u<v\le n} g_{uv}\left(\frac{\beta}{r}q_{uv}\right) \prod_{1\le u\le n} f_u\big(e^{-\frac{\beta}{r}d_u}\big). \end{aligned} \tag{9.29}$$

Here, we have two sources of error: the Trotter error and the error due to using the gate set (9.25). We choose

$$r > 2\beta, \tag{9.30}$$

so that we can implement the product formula using gates from (9.25) with pa-

rameters

$$-\frac{1}{2} < -\frac{\beta}{r} d_u < \frac{1}{2}, \quad 0 < \frac{\beta}{r} q_{uv} < \frac{1}{2}, \quad 0 < \frac{\beta}{r} p_{uv} < \frac{1}{2}. \tag{9.31}$$

Consider the gate sequence

$$\begin{aligned}
&\prod_{1\le u\le n} f_u\big(e^{-\frac{\beta}{r} d_u}\big) \prod_{1\le u<v\le n} g_{uv}\left(\frac{\beta}{r} q_{uv}\right) \prod_{1\le u<v\le n} h_{uv}\left(\frac{\beta}{r} p_{uv}\right) \\
&\quad \cdot \prod_{1\le u<v\le n} h_{uv}\left(\frac{\beta}{r} p_{uv}\right) \prod_{1\le u<v\le n} g_{uv}\left(\frac{\beta}{r} q_{uv}\right) \prod_{1\le u\le n} f_u\big(e^{-\frac{\beta}{r} d_u}\big) \\
&= \prod_{1\le u\le n} e^{-\frac{\beta}{2r} d_u F_u} \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r} q_{uv} \widetilde{\mathscr{G}}_{uv}(\frac{\beta}{r} q_{uv})} \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r} p_{uv} \widetilde{\mathscr{H}}_{uv}(\frac{\beta}{r} p_{uv})} \\
&\quad \cdot \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r} p_{uv} \widetilde{\mathscr{H}}_{uv}(\frac{\beta}{r} p_{uv})} \prod_{1\le u<v\le n} e^{-\frac{\beta}{2r} q_{uv} \widetilde{\mathscr{G}}_{uv}(\frac{\beta}{r} q_{uv})} \prod_{1\le u\le n} e^{-\frac{\beta}{2r} d_u F_u} \\
&= \exp\left(-\frac{\beta}{r} \left(\sum_{1\le u<v\le n} p_{uv} \widetilde{\mathscr{H}}_{uv}\left(\frac{\beta}{r} p_{uv}\right) + \sum_{1\le u<v\le n} q_{uv} \widetilde{\mathscr{G}}_{uv}\left(\frac{\beta}{r} q_{uv}\right) + \sum_{u=1}^{n} d_u F_u \right) \right) \cdot W
\end{aligned} \tag{9.32}$$

that implements the second-order Suzuki formula, where we have applied Theorem 33 in the last line. Since

$$\|F_u\| \le 2, \quad \left\| \widetilde{\mathscr{G}}_{uv}\left(\frac{\beta}{r} q_{uv}\right) \right\| \le 2 + 2\frac{\beta}{r} q_{uv} \le 3, \quad \left\| \widetilde{\mathscr{H}}_{uv}\left(\frac{\beta}{r} p_{uv}\right) \right\| \le 2 + 2\frac{\beta}{r} p_{uv} \le 3, \tag{9.33}$$

the perturbed Hamiltonian satisfies

$$\begin{aligned}
&\sum_{1\le u<v\le n} p_{uv} \left\| \widetilde{\mathscr{H}}_{uv}\left(\frac{\beta}{r} p_{uv}\right) \right\| + \sum_{1\le u<v\le n} q_{uv} \left\| \widetilde{\mathscr{G}}_{uv}\left(\frac{\beta}{r} q_{uv}\right) \right\| + \sum_{u=1}^{n} |d_u| \, \|F_u\| \\
&\le \binom{n}{2} 3 + \binom{n}{2} 3 + 2n \le 3n^2.
\end{aligned} \tag{9.34}$$

We also need to bound nested commutators of Hamiltonian terms with two layers of nesting. This analysis is similar to that for the transverse field Ising model; the resulting scaling is $\mathcal{O}(n^4)$. By Theorem 39, there exists a constant $c > 0$ such that

$$\|W\| \leq \exp\left(\frac{cn^4\beta^3}{r^3} e^{\frac{6n^2\beta}{r}}\right). \tag{9.35}$$

To proceed, we apply Lemma 2 to switch to the interaction picture, giving

$$\exp\left(-\frac{\beta}{r}\left(\sum_{1\leq u<v\leq n} p_{uv}\widetilde{\mathscr{H}}_{uv}\left(\frac{\beta}{r}p_{uv}\right)+\sum_{1\leq u<v\leq n} q_{uv}\widetilde{\mathscr{G}}_{uv}\left(\frac{\beta}{r}q_{uv}\right)+\sum_{u=1}^{n} d_u F_u\right)\right) = e^{-\frac{\beta}{r}H}V, \tag{9.36}$$

where

$$V = \exp_{\mathcal{T}}\left(-\int_0^{\frac{\beta}{r}} \mathrm{d}\tau\; e^{\tau H}\left(\sum_{1\leq u<v\leq n} p_{uv}\widetilde{\mathscr{H}}_{uv}\left(\frac{\beta}{r}p_{uv}\right) + \sum_{1\leq u<v\leq n} q_{uv}\widetilde{\mathscr{G}}_{uv}\left(\frac{\beta}{r}q_{uv}\right) + \sum_{u=1}^{n} d_u F_u - H\right)e^{-\tau H}\right). \tag{9.37}$$

From (9.28),

$$\begin{aligned}
&\left\|\sum_{1\leq u<v\leq n} p_{uv}\widetilde{\mathscr{H}}_{uv}\left(\frac{\beta}{r}p_{uv}\right) + \sum_{1\leq u<v\leq n} q_{uv}\widetilde{\mathscr{G}}_{uv}\left(\frac{\beta}{r}q_{uv}\right) + \sum_{u=1}^{n} d_u F_u - H\right\| \\
&= \left\|\sum_{1\leq u<v\leq n} p_{uv}\frac{2r}{\beta p_{uv}}\mathscr{H}_{uv}\left(\frac{\beta}{r}p_{uv}\right) + \sum_{1\leq u<v\leq n} q_{uv}\frac{2r}{\beta q_{uv}}\mathscr{G}_{uv}\left(\frac{\beta}{r}q_{uv}\right)\right\| \\
&\leq \binom{n}{2}2\frac{\beta}{r}p_{uv} + \binom{n}{2}2\frac{\beta}{r}q_{uv} = 2n^2\frac{\beta}{r},
\end{aligned} \tag{9.38}$$

whereas the original Hamiltonian has spectral norm

$$\begin{aligned}
&\|H\| \\
\leq &\sum_{1\leq u<v\leq n} p_{uv}\,\|-X_uX_v - Y_uY_v\| + \sum_{1\leq u<v\leq n} q_{uv}\,\|-X_uX_v + Y_uY_v\| + \sum_{u=1}^{n} |d_u|\,\|I + Z_u\| \\
\leq &\binom{n}{2}\cdot 2 + \binom{n}{2}\cdot 2 + n\cdot 2 = 2n^2,
\end{aligned} \tag{9.39}$$

so Lemma 4 implies

$$\|V\| \leq \exp\left(\frac{2n^2\beta^2}{r^2} e^{\frac{4n^2\beta}{r}}\right). \tag{9.40}$$

Altogether, we obtain

$$\begin{aligned}
&\prod_{1\leq u\leq n} f_u\big(e^{-\frac{\beta}{r}d_u}\big) \prod_{1\leq u<v\leq n} g_{uv}\left(\frac{\beta}{r}q_{uv}\right) \prod_{1\leq u<v\leq n} h_{uv}\left(\frac{\beta}{r}p_{uv}\right) \\
&\quad\cdot \prod_{1\leq u<v\leq n} h_{uv}\left(\frac{\beta}{r}p_{uv}\right) \prod_{1\leq u<v\leq n} g_{uv}\left(\frac{\beta}{r}q_{uv}\right) \prod_{1\leq u\leq n} f_u\big(e^{-\frac{\beta}{r}d_u}\big) \\
&= e^{-\frac{\beta}{r}H}U,
\end{aligned} \tag{9.41}$$

where the operator $U = VW$ has spectral norm bounded by

$$\|U\| = \|VW\| \leq \exp\left(\frac{2n^2\beta^2}{r^2} e^{\frac{4n^2\beta}{r}} + \frac{cn^4\beta^3}{r^3} e^{\frac{6n^2\beta}{r}}\right) \tag{9.42}$$

for some constant $c > 0$.

The remaining analysis proceeds in a similar way as that of the transverse

field Ising model. We find that each eigenvalue of

$$\left[\prod_{1 \le u \le n} f_u\left(e^{-\frac{\beta}{r} d_u}\right) \prod_{1 \le u < v \le n} g_{uv}\left(\frac{\beta}{r} q_{uv}\right) \prod_{1 \le u < v \le n} h_{uv}\left(\frac{\beta}{r} p_{uv}\right) \right. \\ \left. \cdot \prod_{1 \le u < v \le n} h_{uv}\left(\frac{\beta}{r} p_{uv}\right) \prod_{1 \le u < v \le n} g_{uv}\left(\frac{\beta}{r} q_{uv}\right) \prod_{1 \le u \le n} f_u\left(e^{-\frac{\beta}{r} d_u}\right) \right]^r \tag{9.43}$$

approximates the corresponding eigenvalue of the ideal evolution $e^{-\beta H}$ with a multiplicative factor

$$\|U\|^r \le \exp\left(\frac{2n^2\beta^2}{r} e^{\frac{4n^2\beta}{r}} + \frac{cn^4\beta^3}{r^2} e^{\frac{6n^2\beta}{r}} \right). \tag{9.44}$$

We first set

$$r \ge 12n^2\beta, \tag{9.45}$$

so that

$$\|U\|^r \le \exp\left(\frac{4n^2\beta^2}{r} + \frac{2cn^4\beta^3}{r^2} \right). \tag{9.46}$$

We then choose

$$r = \max\left\{ \frac{8n^2\beta^2}{\epsilon}, \frac{2\sqrt{c}n^2\beta^{3/2}}{\epsilon^{1/2}} \right\} \tag{9.47}$$

to ensure that the multiplicative error is at most ϵ. By (9.30), (9.45), and (9.47), we have

$$r = \mathcal{O}\left(\frac{n^2\beta^2}{\epsilon} \right), \tag{9.48}$$

which gives the total gate complexity [21, Supplementary p. 7]

$$j = 4n^2 r = \mathcal{O}\left(\frac{(1+\beta^2)n^4}{\epsilon}\right). \tag{9.49}$$

The result of [21, Theorem 2] gave a Monte Carlo simulation algorithm for the ferromagnetic quantum spin systems. To improve that result, we also need to estimate the error of partial sequence of the product formula as in [21, Eq. (13)]. This can be done in a similar way as our above analysis. The resulting randomized approximation scheme has runtime

$$\widetilde{\mathcal{O}}\left(\frac{j^{23}}{\epsilon^2}\right) = \widetilde{\mathcal{O}}\left(\frac{n^{92}(1+\beta^{46})}{\epsilon^{25}}\right), \tag{9.50}$$

which improves the runtime of the original Bravyi-Gosset algorithm

$$\widetilde{\mathcal{O}}\left(\frac{n^{115}(1+\beta^{46})}{\epsilon^{25}}\right). \tag{9.51}$$

Chapter 10: Conclusion and future work

In this dissertation, we developed an understanding of quantum simulation algorithms concerning their design, analysis, implementation, and application.

Specifically, in Chapter 2, we discussed time-ordered evolutions and their mathematical properties, introduced common input models (the linear-combination and linear-combination-of-unitaries model) for quantum simulation, and reviewed leading simulation algorithms (the product-formula, Taylor-series, and quantum-signal-processing algorithm).

In Chapter 3, we considered the simulation of a one-dimensional Heisenberg spin model and compared the resource requirements of implementing different quantum simulation algorithms. We found that the quantum-signal-processing approach has the best performance with rigorous accuracy guarantee, whereas the product-formula approach performs significantly better with empirical estimate. We obtained much smaller circuits than those for the simplest classically-infeasible instances of factoring and quantum chemistry, identifying simulation of spin systems as a potential candidate problem for near-term demonstration of quantum simulation.

In Chapter 4, we developed a randomized approach to quantum simulation using product formulas. We randomly permuted the ordering of summands in product-formula simulation and compared this new approach with its deterministic

counterpart. We showed that randomized product formulas are asymptotically advantageous over deterministic formulas and this advantage remains to hold even in practice.

In Chapter 5, we designed a randomized approach to time-dependent Hamiltonian simulation. Specifically, we developed a classical sampling protocol using a probability distribution that biases toward those times at which the instantaneous norm of the Hamiltonian is large. Previous simulation algorithms have complexity that depends on the worst-case instantaneous norm, but our new approach scales with the integral average. Our approach is thus advantageous for Hamiltonians varying significantly with time, as in semi-classical simulations of scattering processes in quantum chemistry.

In Chapter 6, we proposed a general theory for analyzing the error of product formulas (Trotter error). Previous work obtained tight error analysis for certain lower-order product formulas and special systems, such as those with Lie-algebraic structure, but our approach holds in general. We considered Trotter error of various types, including additive error, multiplicative error, and error that appears in the exponent. For each type, we applied the correct order condition to cancel lower-order terms, and represented higher-order ones as explicit nested commutators.

In Chapter 7, we analyzed the performance of product formulas for simulating many concrete systems, including nearest-neighbor lattice systems, electronic structure Hamiltonians, k-local Hamiltonians, rapidly decaying power-law interac-

tions, and clustered Hamiltonians. Applying our theory of Trotter error, we showed that the performance of product formulas can nearly match or even outperform the best previous results in digital quantum simulation. We further numerically benchmarked our analysis, showing that our error bounds also have nearly tight constant prefactors.

In Chapter 8, we developed an algorithmic framework "quantum singular value transformation" based on ideas from quantum simulation. We described the mathematical setting of this framework and proved a theorem that relates the spectra of the operator we want to transform and the one that can be implemented on a quantum computer. As an application, we used this framework to implement principal component regression in machine learning.

In Chapter 9, we considered applications of our Trotter error analysis to quantum Monte Carlo methods. We consider two specific systems: the transverse field Ising model and the ferromagnetic quantum spin systems. For both systems, we showed that previous Monte Carlo simulations can be made more efficient using our tightened analysis.

Beyond those open problems already mentioned in the previous chapters, there are several other questions regarding quantum simulation algorithms that have not been answered in this dissertation.

Beyond the spin model we have considered in Chapter 3, quantum chemistry also provides a natural choice of target system for near-term quantum simulation.

Indeed, simulating chemical systems has industrial relevance in the design and engineering of new pharmaceuticals, catalysts and materials. Many previous approaches to quantum chemistry use product formulas. However, due to the lack of tight error bound, their results often overestimate the gate complexity by several orders of magnitude. It would be interesting to further explore the extent to which the cost of quantum chemistry simulation can be reduced using our analysis in Chapter 6 and Chapter 7. Other systems, such as those in nuclear physics and quantum field theory, are also natural candidates for quantum simulation [95], although they have received far less attention in recent studies. It would be fruitful to estimate the quantum cost of simulating such systems that are otherwise infeasible to simulate on current classical computers.

Chapter 4, Chapter 5 and recent work [25, 84] developed faster algorithms for quantum simulation using classical randomness. These randomized algorithms have better asymptotic dependence on certain parameters and can be advantageous in practice as well. However, due to the use of randomness, these approaches only achieve first-order accuracy. It may then be beneficial to use deterministic methods that are higher-order accurate while retaining advantages of randomized algorithms. Several possible strategies were suggested in quantum chemistry simulation including the coalescing strategy [87, 107] and the divide-and-conquer strategy [54], of which it would be interesting to develop a further understanding.

Finally, one may ask if quantum simulation algorithms could find further

applications, especially to areas of physics beyond quantum computing. One possibility is to apply ideas from quantum simulation to study quantum Zeno effect, a feature of quantum dynamics that has been explored both theoretically and experimentally. Recent work [22] derived a concrete bound for the rate of convergence of quantum Zeno effect, although their bound contains an exponential prefactor that prevents it from being useful in practice. As part of their approach, they considered a decomposition of evolution based on product formulas and that may be made more efficient using ideas from Chapter 6. We hope the results of this dissertation could provide insights to such applications, which would have immediate value even if scalable quantum computers may not be available in the near future.

Bibliography

[1] Dorit Aharonov and Amnon Ta-Shma, *Adiabatic quantum state generation and statistical zero knowledge*, Proceedings of the 35th ACM Symposium on Theory of Computing, pp. 20–29, 2003, arXiv:quant-ph/0301023.

[2] Alán Aspuru-Guzik, Anthony D. Dutoi, Peter J. Love, and Martin Head-Gordon, *Simulated quantum computation of molecular energies*, Science **309** (2005), no. 5741, 1704–1707, arXiv:quant-ph/0604193.

[3] Yosi Atia and Dorit Aharonov, *Fast-forwarding of Hamiltonians and exponentially precise measurements*, Nature communications **8** (2017), no. 1, 1572, arXiv:1610.09619.

[4] Winfried Auzinger and Wolfgang Herfort, *Local error structures and order conditions in terms of Lie elements for exponential splitting schemes*, Opuscula Mathematica **34** (2014), no. 2, 243–255.

[5] Winfried Auzinger, Othmar Koch, and Mechthild Thalhammer, *Defect-based local error estimators for splitting methods, with application to Schrödinger equations, Part ii. Higher-order methods for linear problems*, Journal of Computational and Applied Mathematics **255** (2014), 384–403.

[6] Sheldon Axler, *Linear algebra done right*, Springer, 2015.

[7] Ryan Babbush, Craig Gidney, Dominic W. Berry, Nathan Wiebe, Jarrod McClean, Alexandru Paler, Austin Fowler, and Hartmut Neven, *Encoding electronic spectra in quantum circuits with linear T complexity*, Physical Review X **8** (2018), 041015, arXiv:1805.03662.

[8] Ryan Babbush, Jarrod McClean, Dave Wecker, Alán Aspuru-Guzik, and Nathan Wiebe, *Chemical basis of Trotter-Suzuki errors in quantum chemistry simulation*, Physical Review A **91** (2015), 022311, arXiv:1410.8159.

[9] Ryan Babbush, Nathan Wiebe, Jarrod McClean, James McClain, Hartmut Neven, and Garnet Kin-Lic Chan, *Low-depth quantum simulation of materials*, Physical Review X **8** (2018), 011044, arXiv:1706.00023.

[10] Dave Bacon, *Quantum computing*, 2006, https://courses.cs.washington.edu/courses/cse599d/06wi/.

[11] Dominic W. Berry, *High-order quantum algorithm for solving linear differential equations*, Journal of Physics A: Mathematical and Theoretical **47** (2014), no. 10, 105301, arXiv:1010.2745.

[12] Dominic W. Berry, Graeme Ahokas, Richard Cleve, and Barry C. Sanders, *Efficient quantum algorithms for simulating sparse Hamiltonians*, Communications in Mathematical Physics **270** (2007), no. 2, 359–371, arXiv:quant-ph/0508139.

[13] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma, *Exponential improvement in precision for simulating sparse Hamiltonians*, Proceedings of the 46th Annual ACM Symposium on Theory of Computing, pp. 283–292, 2014, arXiv:1312.1414.

[14] 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** (2015), no. 9, 090502, arXiv:1412.4687.

[15] Dominic W. Berry, Andrew M. Childs, and Robin Kothari, *Hamiltonian simulation with nearly optimal dependence on all parameters*, Proceedings of the 56th IEEE Symposium on Foundations of Computer Science, pp. 792–809, 2015, arXiv:1501.01715.

[16] Dominic W. Berry, Andrew M. Childs, Yuan Su, Xin Wang, and Nathan Wiebe, *Time-dependent Hamiltonian simulation with L^1-norm scaling*, Quantum **4** (2020), 254, arXiv:1906.07115.

[17] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd, *Quantum machine learning*, Nature **549** (2017), 195–202, arXiv:1611.09347.

[18] Sergio Blanes and Fernando Casas, *A concise introduction to geometric numerical integration*, Chapman and Hall/CRC, 2016.

[19] Fernando G. S. L. Brandao and Krysta M. Svore, *Quantum speed-ups for solving semidefinite programs*, Proceedings of the 58th IEEE Symposium on Foundations of Computer Science, pp. 415–426, 2017, arXiv:1609.05537.

[20] Sergey Bravyi, *Monte carlo simulation of stoquastic Hamiltonians*, Quantum Information and Computation **15** (2015), no. 13-14, 1122–1140, arXiv:1402.2295.

[21] Sergey Bravyi and David Gosset, *Polynomial-time classical simulation of quantum ferromagnets*, Physical Review Letters **119** (2017), 100503, arXiv:1612.05602.

[22] Daniel Burgarth, Paolo Facchi, Giovanni Gramegna, and Saverio Pascazio, *Generalized product formulas and quantum control*, Journal of Physics A: Mathematical and Theoretical **52** (2019), no. 43, 435301, arXiv:1906.04498.

[23] Laurie J. Butler, *Chemical reaction dynamics beyond the Born-Oppenheimer approximation*, Annual Review of Physical Chemistry **49** (1998), no. 1, 125–171.

[24] Earl Campbell, *Shorter gate sequences for quantum computing by mixing unitaries*, Physical Review A **95** (2017), 042306, arXiv:1612.02689.

[25] Earl Campbell, *Random compiler for fast Hamiltonian simulation*, Physical Review Letters **123** (2019), 070503, arXiv:1811.08017.

[26] 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** (2019), no. 19, 10856–10915, arXiv:1812.09976.

[27] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery, *The power of block-encoded matrix powers: Improved regression techniques via faster Hamiltonian simulation*, 46th International Colloquium on Automata, Languages, and Programming, vol. 132, pp. 33:1–33:14, 2019, arXiv:1804.01973.

[28] Yu Chen, C. Neill, P. Roushan, N. Leung, M. Fang, R. Barends, J. Kelly, B. Campbell, Z. Chen, B. Chiaro, A. Dunsworth, E. Jeffrey, A. Megrant, J.Y. Mutus, P.J.J. O'Malley, C.M. Quintana, D. Sank, A. Vainsencher, J. Wenner, T.C. White, Michael R. Geller, A.N. Cleland, and John M. Martinis, *Qubit architecture with high coherence and fast tunable coupling*, Physical Review Letters **113** (2014), 220502, arXiv:1402.7367.

[29] Andrew M. Childs, *On the relationship between continuous- and discrete-time quantum walk*, Communications in Mathematical Physics **294** (2010), no. 2, 581–603, arXiv:0810.0312.

[30] Andrew M. Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A. Spielman, *Exponential algorithmic speedup by quantum walk*, Proceedings of the 35th ACM Symposium on Theory of Computing, pp. 59–68, 2003, arXiv:quant-ph/0209131.

[31] Andrew M. Childs and Robin Kothari, *Limitations on the simulation of non-sparse Hamiltonians*, Quantum Information and Computation **10** (2010), no. 7-8, 669–684, arXiv:0908.4398.

[32] Andrew M. Childs, Robin Kothari, and Rolando D. Somma, *Quantum algorithm for systems of linear equations with exponentially improved dependence on precision*, SIAM Journal on Computing **46** (2017), no. 6, 1920–1950, arXiv:1511.02306.

[33] Andrew M. Childs, Dmitri Maslov, Yunseong Nam, Neil J. Ross, and Yuan Su, 2017, https://github.com/njross/simcount.

[34] 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** (2018), no. 38, 9456–9461, arXiv:1711.10980.

[35] Andrew M. Childs, Aaron Ostrander, and Yuan Su, *Faster quantum simulation by randomization*, Quantum **3** (2019), 182, arXiv:1805.08385.

[36] Andrew M. Childs and Yuan Su, *Nearly optimal lattice simulation by product formulas*, Physical Review Letters **123** (2019), 050503, arXiv:1901.00564.

[37] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu, *A theory of Trotter error*, 2019, arXiv:1912.08854.

[38] Anirban Narayan Chowdhury and Rolando D. Somma, *Quantum algorithms for Gibbs sampling and hitting-time estimation*, Quantum Information and Computation **17** (2017), no. 1-2, 41–64, arXiv:1603.02940.

[39] Ronald de Wolf, *The potential impact of quantum computers on society*, Ethics and Information Technology **19** (2017), no. 4, 271–276, arXiv:1712.05380.

[40] Shantanu Debnath, Norbert M. Linke, Caroline Figgatt, Kevin A Landsman, Kevin Wright, and Christopher Monroe, *Demonstration of a small programmable quantum computer with atomic qubits*, Nature **536** (2016), 63–66, arXiv:1603.04512.

[41] Stéphane Descombes and Mechthild Thalhammer, *An exact local error representation of exponential operator splitting methods for evolutionary problems and applications to linear Schrödinger equations in the semi-classical regime*, BIT Numerical Mathematics **50** (2010), 729–749.

[42] John Day Dollard and Charles N. Friedman, *Product integration with application to differential equations*, Cambridge University Press, 1984.

[43] Stanley C. Eisenstat and Ilse C. F. Ipsen, *Relative perturbation techniques for singular value problems*, SIAM Journal on Numerical Analysis **32** (1995), no. 6, 1972–1988.

[44] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann, *A quantum algorithm for the Hamiltonian NAND tree*, Theory of Computing **4** (2008), no. 1, 169–190, quant-ph/0702144.

[45] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda, *A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem*, Science **292** (2001), no. 5516, 472–475, arXiv:quant-ph/0104129.

[46] Andrew J. Ferris, *Fourier transform for fermionic systems and the spectral tensor network*, Physical Review Letters **113** (2014), 010401, arXiv:1310.7605.

[47] Richard P. Feynman, *Simulating physics with computers*, International Journal of Theoretical Physics **21** (1982), no. 6-7, 467–488.

[48] Roy Frostig, Cameron Musco, Christopher Musco, and Aaron Sidford, *Principal component projection without principal component analysis*, Proceedings of the 33rd International Conference on Machine Learning, pp. 2349–2357, 2016, arXiv:1602.06872.

[49] Xun Gao, Sheng-Tao Wang, and Lu-Ming Duan, *Quantum supremacy for simulating a translation-invariant Ising spin model*, Physical Review Letters **118** (2017), 040502, arXiv:1607.04947.

[50] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe, *Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics*, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, pp. 193–204, ACM, 2019, arXiv:1806.01838.

[51] Alexander S. Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron, *Quipper: A scalable quantum programming language*, ACM SIGPLAN Notices **48** (2013), no. 6, 333–342, arXiv:1304.3390.

[52] Lov K. Grover, *A fast quantum mechanical algorithm for database search*, Proceedings of the 28th ACM Symposium on Theory of Computing, pp. 212–219, ACM, 1996.

[53] Jeongwan Haah, Matthew B. Hastings, Robin Kothari, and Guang Hao Low, *Quantum algorithm for simulating real time evolution of lattice Hamiltonians*, Proceedings of the 59th IEEE Symposium on Foundations of Computer Science, pp. 350–360, Oct 2018, arXiv:1801.03922.

[54] Stuart Hadfield and Anargyros Papageorgiou, *Divide and conquer approach to quantum Hamiltonian simulation*, New Journal of Physics **20** (2018), no. 4, 043003.

[55] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd, *Quantum algorithm for linear systems of equations*, Physical Review Letters **103** (2009), no. 15, 150502, arXiv:0811.3171.

[56] Matthew B. Hastings, *Turning gate synthesis errors into incoherent errors*, Quantum Information and Computation **17** (2017), no. 5-6, 488–494, arXiv:1612.01011.

[57] Trygve Helgaker, Poul Jorgensen, and Jeppe Olsen, *Molecular electronic-structure theory*, John Wiley & Sons, 2014.

[58] Markus Heyl, Philipp Hauke, and Peter Zoller, *Quantum localization bounds Trotter errors in digital quantum simulation*, Science Advances **5** (2019), no. 4, arXiv:1806.11123.

[59] Milan Holzäpfel and Martin B. Plenio, *Efficient certification and simulation of local quantum many-body Hamiltonians*, 2017, arXiv:1712.04396.

[60] Roger A. Horn and Charles R. Johnson, *Matrix analysis*, Cambridge university press, 2012.

[61] J. Huyghebaert and H. De Raedt, *Product formula methods for time-dependent Schrödinger problems*, Journal of Physics A: Mathematical and General **23** (1990), no. 24, 5777–5793.

[62] IBM, *The Quantum Experience*, http://www.research.ibm.com/ibm-q/, 2016.

[63] Ilse C. F. Ipsen, *Relative perturbation results for matrix eigenvalues and singular values*, Acta Numerica **7** (1998), 151201.

[64] Stephen P. Jordan, Keith S. M. Lee, and John Preskill, *Quantum algorithms for quantum field theories*, Science **336** (2012), no. 6085, 1130–1133, arXiv:1111.3633.

[65] Mária Kieferová, Artur Scherer, and Dominic Berry, *Simulating the dynamics of time-dependent Hamiltonians with a truncated Dyson series*, Physical Review A **99** (2019), 042314, arXiv:1805.00582.

[66] Ian D. Kivlichan, Craig Gidney, Dominic W. Berry, Nathan Wiebe, Jarrod McClean, Wei Sun, Zhang Jiang, Nicholas Rubin, Austin Fowler, Alán Aspuru-Guzik, Ryan Babbush, and Hartmut Neven, *Improved fault-tolerant*

quantum simulation of condensed-phase correlated electrons via Trotterization, 2019, arXiv:1902.10673.

[67] Ian D. Kivlichan, Jarrod McClean, Nathan Wiebe, Craig Gidney, Alán Aspuru-Guzik, Garnet Kin-Lic Chan, and Ryan Babbush, *Quantum simulation of electronic structure with linear depth and connectivity*, Physical Review Letters **120** (2018), 110501, arXiv:1711.04789.

[68] Anthony W. Knapp, *Basic real analysis*, Birkhëuser, 2005.

[69] Samuel A. Kutin, *Shor's algorithm on a nearest-neighbor machine*, arXiv:quant-ph/0609001.

[70] Seth Lloyd, *Universal quantum simulators*, Science **273** (1996), no. 5278, 1073–1078.

[71] Guang Hao Low, *Hamiltonian simulation with nearly optimal dependence on spectral norm*, Proceedings of the 51th ACM Symposium on Theory of Computing, pp. 491–502, ACM, 2019, arXiv:1807.03967.

[72] Guang Hao Low and Isaac L. Chuang, *Hamiltonian simulation by uniform spectral amplification*, 2017, arXiv:1707.05391.

[73] Guang Hao Low and Isaac L. Chuang, *Optimal Hamiltonian simulation by quantum signal processing*, Physical Review Letters **118** (2017), 010501, arXiv:1606.02685.

[74] Guang Hao Low and Isaac L. Chuang, *Hamiltonian simulation by qubitization*, Quantum **3** (2019), 163, arXiv:1610.06546.

[75] Guang Hao Low, Vadym Kliuchnikov, and Nathan Wiebe, *Well-conditioned multiproduct Hamiltonian simulation*, 2019, arXiv:1907.11679.

[76] Guang Hao Low and Nathan Wiebe, *Hamiltonian simulation in the interaction picture*, 2018, arXiv:1805.00675.

[77] Guang Hao Low, Theodore J. Yoder, and Isaac L. Chuang, *Methodology of resonant equiangular composite quantum gates*, Physical Review X **6** (2016), 041067, arXiv:1603.03996.

[78] David J. Luitz, Nicolas Laflorencie, and Fabien Alet, *Many-body localization edge in the random-field Heisenberg chain*, Physical Review B **91** (2015), no. 8, 081103, arXiv:1411.0660.

[79] Chris Marriott and John Watrous, *Quantum Arthur–Merlin games*, Computational Complexity **14** (2005), no. 2, 122–152, arXiv:cs/0506068.

[80] Dmitri Maslov, *Optimal and asymptotically optimal NCT reversible circuits by the gate types*, Quantum Information and Computation **16** (2016), no. 13-14, 1096–1112, arXiv:1602.02627.

[81] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C. Benjamin, and Xiao Yuan, *Quantum computational chemistry*, Reviews of Modern Physics **92** (2020), 015003, arXiv:1808.10402.

[82] Yunseong Nam, Neil J. Ross, Yuan Su, Andrew M. Childs, and Dmitri Maslov, *Automated optimization of large-scale quantum circuits with continuous parameters*, npj Quantum Information **4** (2018), 23, arXiv:1710.07345.

[83] Michael A. Nielsen and Isaac L. Chuang, *Quantum computation and quantum information*, Cambridge University Press, 2010.

[84] Yingkai Ouyang, David R. White, and Earl T. Campbell, *Compilation by stochastic Hamiltonian sparsification*, Quantum **4** (2020), 235, arXiv:1910.06255.

[85] Shengshi Pang and Andrew N. Jordan, *Optimal adaptive control for quantum metrology with time-dependent Hamiltonians*, Nature Communications **8** (2017), 14695, arXiv:1606.02166.

[86] Tianyi Peng, Aram Harrow, Maris Ozols, and Xiaodi Wu, *Simulating large quantum circuits on a small quantum computer*, 2019, arXiv:1904.00102.

[87] David Poulin, Matthew B. Hastings, Dave Wecker, Nathan Wiebe, Andrew C. Doherty, and Matthias Troyer, *The Trotter step size required for accurate quantum simulation of quantum chemistry*, Quantum Information and Computation **15** (2015), no. 5-6, 361–384, arXiv:1406.4920.

[88] David Poulin, Angie Qarry, Rolando D. Somma, and Frank Verstraete, *Quantum simulation of time-dependent Hamiltonians and the convenient illusion of Hilbert space*, Physical Review Letters **106** (2011), no. 17, 170501, arXiv:1102.1360.

[89] David Poulin and Pawel Wocjan, *Preparing ground states of quantum many-body systems on a quantum computer*, Physical Review Letters **102** (2009), 130503, arXiv:0809.2705.

[90] Markus Reiher, Nathan Wiebe, Krysta M. Svore, Dave Wecker, and Matthias Troyer, *Elucidating reaction mechanisms on quantum computers*, Proceedings of the National Academy of Sciences **114** (2017), no. 29, 7555–7560, arXiv:1605.03590.

[91] Steven Roman, *Linear algebra done right*, Springer, 2008.

[92] Neil J. Ross and Peter Selinger, *Optimal ancilla-free Clifford+T approximation of z-rotations*, Quantum Information and Computation **16** (2016), no. 11-12, 901–953, arXiv:1403.2975.

[93] Michael Schreiber, Sean S. Hodgman, Pranjal Bordia, Henrik P. Lüschen, Mark H. Fischer, Ronen Vosk, Ehud Altman, Ulrich Schneider, and Immanuel Bloch, *Observation of many-body localization of interacting fermions in a quasirandom optical lattice*, Science **349** (2015), no. 6250, 842–845, arXiv:1501.05661.

[94] M. Serbyn, M. Knap, S. Gopalakrishnan, Z. Papić, N. Y. Yao, C. R. Laumann, D. A. Abanin, M. D. Lukin, and E. A. Demler, *Interferometric probes of many-body localization*, Physical Review Letters **113** (2014), no. 14, 147204, arXiv:1403.0693.

[95] Alexander F. Shaw, Pavel Lougovski, Jesse R. Stryker, and Nathan Wiebe, *Quantum algorithms for simulating the lattice Schwinger model*, 2020, arXiv:2002.11146.

[96] Lukas M. Sieberer, Tobias Olsacher, Andreas Elben, Markus Heyl, Philipp Hauke, Fritz Haake, and Peter Zoller, *Digital quantum simulation, Trotter errors, and quantum chaos of the kicked top*, npj Quantum Information **5** (2019), no. 1, 1–11, arXiv:1812.05876.

[97] Jacob Smith, Aaron Lee, Philip Richerme, Brian Neyenhuis, Paul W. Hess, Philipp Hauke, Markus Heyl, David A. Huse, and Christopher Monroe, *Many-body localization in a quantum simulator with programmable random disorder*, Nature Physics **12** (2016), 907–911, arXiv:1508.07026.

[98] Rolando D. Somma, *A Trotter-Suzuki approximation for Lie groups with applications to Hamiltonian simulation*, Journal of Mathematical Physics **57** (2016), 062202, arXiv:1512.03416.

[99] Masuo Suzuki, *Decomposition formulas of exponential operators and Lie exponentials with some applications to quantum mechanics and statistical physics*, Journal of Mathematical Physics **26** (1985), no. 4, 601–612.

[100] Masuo Suzuki, *General theory of fractal path integrals with applications to many-body theories and statistical physics*, Journal of Mathematical Physics **32** (1991), no. 2, 400–407.

[101] Mario Szegedy, *Quantum speed-up of markov chain based algorithms*, Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, pp. 32–41, Oct 2004, arXiv:quant-ph/0401053.

[102] Minh C. Tran, Andrew Y. Guo, Yuan Su, James R. Garrison, Zachary Eldredge, Michael Foss-Feig, Andrew M. Childs, and Alexey V. Gorshkov, *Locality and digital quantum simulation of power-law interactions*, Physical Review X **9** (2019), 031006, arXiv:1808.05225.

[103] F. Verstraete and J. I. Cirac, *Mapping local Hamiltonians of fermions to local Hamiltonians of spins*, Journal of Statistical Mechanics **2005** (2005), no. 09, P09012, arXiv:cond-mat/0508353.

[104] John Watrous, *Introduction to quantum computing*, 2006, https://cs.uwaterloo.ca/~watrous/LectureNotes.html.

[105] John Watrous, *Simpler semidefinite programs for completely bounded norms*, Chicago Journal of Theoretical Computer Science **2013** (2013), no. 8, arXiv:1207.5726.

[106] John Watrous, *The theory of quantum information*, Cambridge University Press, 2018.

[107] Dave Wecker, Bela Bauer, Bryan K. Clark, Matthew B. Hastings, and Matthias Troyer, *Gate count estimates for performing quantum chemistry on small quantum computers*, Physical Review A **90** (2014), 022305, arXiv:1312.1695.

[108] Dave Wecker, Matthew B Hastings, Nathan Wiebe, Bryan K Clark, Chetan Nayak, and Matthias Troyer, *Solving strongly correlated electron models on a quantum computer*, Physical Review A **92** (2015), no. 6, 062318, arXiv:1506.05135.

[109] James D. Whitfield, Jacob Biamonte, and Alán Aspuru-Guzik, *Simulation of electronic structure Hamiltonians using quantum computers*, Molecular Physics **109** (2011), no. 5, 735–750, arXiv:1001.3855.

[110] Nathan Wiebe, Dominic Berry, Peter Høyer, and Barry C Sanders, *Higher order decompositions of ordered operator exponentials*, Journal of Physics A **43** (2010), no. 6, 065203, arXiv:0812.0562.

[111] Nathan Wiebe, Daniel Braun, and Seth Lloyd, *Quantum algorithm for data fitting*, Physical Review Letters **109** (2012), 050505, arXiv:1204.5242.

[112] Mark M. Wilde, *Quantum information theory*, Cambridge University Press, 2017.

[113] Chi Zhang, *Randomized algorithms for Hamiltonian simulation*, Monte Carlo and Quasi-Monte Carlo Methods (L. Plaskota and J. Woźniakowski, eds.), vol. 23, pp. 709–719, Springer, 2010, arXiv:0910.4145.