Preparing Thermal States on a Digital Quantum Computer

by

Matthew Hagan

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Department of Physics
University of Toronto

Preparing Thermal States on a Digital Quantum Computer

Matthew Hagan
Doctor of Philosophy

Department of Physics
University of Toronto
2025

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.

This thesis is dedicated to...

Acknowledgements

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The goal of this thesis is to serve as a blueprint for creating quantum channels that can prepare thermal states of arbitrary systems. This framework was primarily created as an algorithmic process to prepare input states of the form $\rho(\beta) = \frac{e^{-\beta H}}{\text{tr}}\left(e^{-\beta H}\right)$, known as Gibbs, Boltzmann, or thermal states, for simulations on a digital, fault-tolerant quantum computer. As $\rho(\beta)$ approaches the ground state as the inverse temperature β diverges, meaning Gibbs states serve as useful proxies for problems in which ground states dictate features of dynamics, such as the electron wavefunction in banded condensed matter systems or in chemical scenarios. One of the key features of our algorithm is the addition of only one single extra qubit outside of those needed to store the state of quantum system being simulated. This may seem like a minor technical achievement, given the existence of Gibbs samplers that also have only one extra qubit or at worst a constant number of qubits overhead, but the way in which this single qubit is utilized in our algorithm highlights the connections between our channel as an algorithmic tool to prepare states of interest and the model of thermalization that we believe the physical world may actually follow. The rest of this introduction is to provide context for how the technical results contained in later chapters of this thesis contribute to the growing interplay between Physics and Theoretical Computer Science within the realm of Quantum Computing.

The easiest way to grasp the context of this thesis is to first understand classical thermal states. In classical mechanics we typically have access to a Hamiltonian H that is a function on phase space (x, p) to the reals $\mathbb{R}$. We can turn this function into a probability distribution via the canonical ensemble $p_\beta(x, p) = \frac{e^{-\beta H(x,p)}}{\int dx dp e^{-\beta H(x,p)}}$, which can be thought of as the "density" of particles near a particular (x, p) in phase space for a thermodynamically large collection of non-interacting particles under the same Hamiltonian H. The inverse temperature β, typically taken to be $\frac{1}{k_\beta T}$ where k_β is Boltzmann's constant and T the temperature, plays an important role in shaping the distribution $p_{\beta(x,p)}$. For example, in the $\beta \to \infty$ the distribution becomes concentrated at the minimum energy points of $H(x, p)$, which correspond to points with zero momentum $p = 0$ and are minimums of the potential energy $V(x)$. This shows that the problem of preparing classical thermal states somehow contains'' the problem of function optimization, and in fact being able to sample from classical thermal states for arbitrarily large β allows us to approximate the partition function $\mathcal{Z} = \int e^{-\beta H(x,p)} dxdp$, which is known to be a #P-Hard computational problem.

In order to understand when classical systems can reach thermal equilibrium we need to understand where does the concept of temperature come from? We can use the following thought experiment of two systems A and B that are completely isolated from their surroundings, each initially has some internal energy U_A and U_B. We could define entropies σ_A and σ_B from the distribution of the energy of each particle, averaged over some time window, and declare that an equilibrium has been reached between the two systems when the total entropy is constant

$$d\sigma = d\sigma_A + d\sigma_B = 0. \tag{1.1}$$

Then we know that the total energy of the two systems must remain constant, due to thermal isolation, so $dU = dU_A + dU_B = 0$. However, in this scenario the energy of the system is only a function of the entropy, so we have

$$dU = \frac{\partial U_A}{\partial \sigma_A} d\sigma_A + \frac{\partial U_B}{\partial \sigma_B} d\sigma_B = \left(\frac{\partial U_A}{\partial \sigma_A} - \frac{\partial U_B}{\partial \sigma_B} \right) d\sigma_A = 0. \tag{1.2}$$

This implies that when the two systems are in equilibrium $\frac{\partial U_A}{\partial \sigma_A} - \frac{\partial U_B}{\partial \sigma_B} = 0$ and the property $\frac{\partial U_A}{\partial \sigma_A}$ is equal to that of B. This is a good notion of temperature, so we define $T = 1/\frac{\partial U}{\partial \sigma}$.

The rationale for introducing this experiment is two-fold. On the first, it gives us a very rigorous definition of temperature by isolating exactly the condition meant behind "thermal equilibrium". We find that we can still define temperature even if we are not sure of how the two systems exchange energy. All we really need to know is that the systems are completely isolated and that energy is exchanged *somehow*, and that this energy exchange is captured by the entropy. The second main point of this experiment is that it gives a very concrete way to thermalize a system to some temperature T: if you have another system at that temperature just put the two into contact! Eventually they will equilibrate to some intermediate temperature and the process can be repeated.

Although the above result tells us that two systems in thermal contact with each other will eventually reach thermal equilibrium it doesn't exactly tell us *how* it does so. Empirically, we have found a few different ways in which two systems can exchange heat:

- They can trade photons at various wavelengths, described by the Plank law, which then get "absorbed" by the various parties. This is how the earth reaches a thermal equilibrium with the sun and the vacuum of space.
- The nuclei of molecules can vibrate, which can then cause nearby nuclei and electrons to oscillate as well through Coulomb's law. This is the process of conduction, and can be extended to rigid crystals via phonon theory.
- Convection allows for hot parts of a liquid or gas to mix with colder parts. This brings the two extremal parts of a system in closer contact and allows for their temperatures to average out quicker.

These three mechanisms are typically sufficient for classical systems to equilibrate. Typically rates of heat exchange are measured empirically and the random microscopic effects are essentially "averaged out" at the macroscopic into a single coefficient of heat transfer.

If one has complete control over some system, such as a refrigerator, and can prepare it in a thermal state of temperature T_{cold}, we can then utilize this to cool down other systems via equilibration. Given our understanding that preparing thermal states is an incredibly challenging problem in the worst case, this raises the question: can we mimick this cooling process algorithmically on digital computers to sample from thermal distributions? If nature is able to cool systems down efficiently, we should be able to do so as well on computer simulations.

When trying to simulate a classical system we can typically compute the energy of the system $H(x, p)$ given a configuration of position and momenta. For example if we are trying to prepare the thermal state for a gas of non-interacting molecules with no background potential then the energy is just $\sum_i p_i^2/2m$. If we encode the positions into standardized 64-bit floating point numbers, computing the energy can be done in order $O(N)$ time, where N is the number of particles simulated. This is referred to as having oracle access to H, once we have a collection

(x, p) we can then compute H as a function call. The challenging problem is to then take this oracle H and output a list of samples $\{s_1, s_2, ..., s_S\}$, where s_i is a pair of position and momenta $s_i = (x_i, p_i)$, that replicate the thermal statistics. We want to use these samples to compute difficult thermal quantities, given some observable $O(x, p)$ (e.g. the velocity of a particle $v = |\vec{p}|$) we would like the following to be as close to each other as possible

$$\frac{1}{S}\sum_{i=1}^{S} O(x_i, p_i)\frac{e^{-\beta H(x_i,p_i)}}{\sum_{j=1}^{S} e^{-\beta H(x_j,p_j)}} \approx \int dxdp\; O(x,p)\frac{e^{-\beta H(x,p)}}{\int e^{-\beta H(x',p')}dx'dp'}. \tag{1.3}$$

This is known as Monte Carlo integration, as the integral on the right is replaced by the random process on the left.

The earliest technique for attacking such problems computationally was the Metropolis-Hastings algorithm. This generated samples $\{s_i\}$ using a two-step process that was repeated over and over. Starting with some random initial sample s_0 one then generates a proposed sample $\tilde{s_1}$ using a random transition $T(x'|\, x)$ over the state space. For example, if the system of interest is a collection of spins $\vec{\mu}_j$ then we can pick one spin uniformly and then randomly rotate it. Once we have the proposed new sample $\tilde{s_1}$ we decide to accept it, in which case $s_1 = \tilde{s_1}$ or reject it, in which case we go back to the previous point and set $s_1 = s_0$, with the randomized filter

$$\Pr[s_i = \tilde{s_i}] = \min\left(1, \frac{e^{-\beta H(\tilde{s_i})}/\mathcal{Z}}{e^{-\beta H(s_{i-1})}/\mathcal{Z}}\right) = \min\bigl(1, e^{-\beta(H(\tilde{s_i})-H(s_{i-1}))}\bigr). \tag{1.4}$$

This is known as the Metropolis filter and one key insight is that it only depends on the difference in energy between the previous state and the proposed state, the dependence on the partition function vanishes when taking the ratio. Since the initial sample can be random and the transition step is also randomized, this leads to an efficiently computable algorithm for generating new samples.

The last remaining question is if these samples are actually representative of the thermal distribution $p_\beta(x, p)$. So long as the transition matrix is ergodic, meaning any starting state (x', p') has a finite expected hitting time to reach any other arbitrary state (x'', p''), then the thermal distribution $p_\beta(x, p)$ satisfies a condition known as Detailed Balance which then guarantees convergence to $p_\beta(x, p)$. As the above process is a Markov chain, since the distribution over the next sample only depends on the state of the current sample, we can define the Markov transition probability (which is different from the state transition probabilities) as $\Pr[(x, p) \to$

$(x', p')]$ and we say that a state probability distribution π satisfies Detailed Balance if the following equation holds

$$\Pr[(x_1, p_1) \to (x_2, p_2)]\pi(x_1, p_1) = \Pr[(x_2, p_2) \to (x_1, p_1)]\pi(x_2, p_2). \tag{1.5}$$

This condition, along with ergodicity, is sufficient to prove that the probability s_i, for $i \gg 1$, will be distributed according to π. The amount of samples needed to converge to π is an incredibly difficult question to answer and the subject of much study on Markov chains.

One of the most recent improvements to this algorithm came about in the late $1980'$s and was formalized in Radford Neal's thesis in the Computer Science Department here at the University of Toronto [1]. This new algorithm, now called Hamiltonian Monte Carlo (HMC) but previously known as Hybrid Monte Carlo, modified the existing Metropolis-Hastings algorithm by changing the state transition function. Instead of choosing the next proposed sample randomly, in HMC the momentum variable is chosen from a Gaussian distribution with variance proportional to $1/\beta$ and then the time dynamics generated by $H(x, p)$ are used to propose a new sample. Since time evolution does not change the energy, the Metropolis filter then accepts every sample in the limit of perfect numerical integration of Hamilton's equations of motion. This technique allows for much higher dimensional state spaces to be explored and empirically leads to less correlated samples.

We introduce HMC for the conceptual changes it makes to the original Metropolis-Hastings method. In Metropolis-Hastings the filter step is really what fixes the distribution to mimic the Gibbs distribution and the transition step is simply to guarantee ergodicity. The filtration step in this sense is rather artificial and more computational in spirit. In HMC the filtration step is virtually eliminated by instead utilizing the dynamics provided by nature. We are only able to take advantage of this because in classical mechanics the position and momentum variables "commute", meaning that the thermal distribution is really a product of distributions over position and momenta as

$$e^{-\beta H(x,p)} = e^{-\beta \frac{p^2}{2m} - \beta V(x)} = e^{-\beta \frac{p^2}{2m}} \cdot e^{-\beta V(x)}, \tag{1.6}$$

and the partition function also splits as a product $\mathcal{Z} = \mathcal{Z}_x \cdot \mathcal{Z}_p$. In a sense, HMC is using time dynamics to provide equilibration not between multiple particles but instead between momenta and position, and due to the simple nature of momentum thermal states as Gaussian random variables we can prepare these states at arbitrary temperatures relatively easy.

This crucial step of commutativity of x and p proves to be a difficult obstactle to overcome when extending HMC to a quantum mechanical setting. If one takes a naïve adoption to a continuous, single variable quantum Hamiltonian $H(\hat{x}, \hat{p})$ and utilizes Gaussian distributed momentum *kicks* $e^{ip_{\text{kick}}\hat{x}}|\psi\rangle$, where p_{kick} is Gaussian and $\hat{x}$ is the position operator that generates momentum translations, one can then show that the maximally mixed state is the unique fixed point of the dynamics. This says that our model of a thermalizing environment as providing random momentum shifts is fundamentally **wrong**. Instead, one needs to come up with a simpler system that can be prepared in the thermal state at the desired temperature and use that to cool (or heat) the system to the target temperature.

The smaller environment that we ended up landing on is a single additional two level system, or qubit. The next difficulty is then to choose an interaction between the qubit and the system of interest that leads to thermalization. When looking to physics for inspiration, the situation for thermalization is much less clear in the quantum setting compared to the clear picture we have classically. Even determining a model for equilibrium is not as clear cut. If we look at a closed setting with two Hilbert spaces $\mathcal{H}_A \otimes \mathcal{H}_B$, similar to our classical scenario, one needs to determine the interaction model between the two halves. Further, the energy of each half is no longer a single function but rather a random variable dependent on the density matrix ρ. Even computing derivatives of the mean energy is difficult as one needs an expression for ρ to compute the von Neumann entropy $-\operatorname{Tr}[\rho \log \rho]$.

There are two main approaches physicists have taken to avoid these complications: one involves a conjecture on large, closed quantum systems and the other involves modelling open quantum systems. The first involves a conjecture known as the Eigenstate Thermalization Hypothesis (ETH) [2] and outlines conditions in which a very large closed quantum system may replicate thermal expectations for *local* observables on single particles. This framework has many connections to quantum chaos and has been proven to hold in such chaotic systems, but remains unproven for generic quantum systems, hence the "hypothesis" in the name. The second connection involves studying the effects of a quantum environment on a quantum system and essentially ignoring the state of the environment. By making several assumptions, such as an infinitely large environment, weak coupling, and an assumption known as Markovianity (where the system and environment remain in product states after each interaction), one can show that the thermal state is the unique fixed point for generic open systems. This is done using the Davies' Generators [3] that give rise to a Linbladian evolution on the system with the desired fixed point. However, there exist many scenarios in nature where these assumptions are not met, such as finite sized environments that retain some correlations with the system and strong coupling situations. Finding Linblad evolutions that allow for generic thermalization for arbitrary system and environment pairs remains an open question. However, we do note that a flurry of quantum algorithms have been developed in recent years that are modeled off of these Linblad dynamics [4–7].

The model for thermalization that we propose is instead rooted in the concepts developed by Hamiltonian Monte Carlo: prepare an additional register in the desired thermal state, let time evolution mix the two constituents, and then refresh the controllable register. As mentioned before, we will make do with just a single qubit ancilla. In order for this process to work there are two difficulties that must be overcome in the quantum mechanical setting: the first is that a Hamiltonian must be chosen for the single ancilla, which amounts to choosing an energy gap we denote γ, and the second is that an interaction term must be chosen between the ancilla and the system. One of the main contributions of this thesis is a rigorous proof that choosing from a suitably random ensemble of interactions is sufficient for thermalization.

This model turns out to have many beneficial properties that existing methods do not have. The first is that this model is very straightforward to implement on a quantum computer using existing algorithms. Evolution by a time independent Hamiltonian is the most complicated subroutine necessary. We will develop these primitives from scratch, showing how simple product formulas can be implemented at the gate level by utilizing a Pauli decomposition of the Hamiltonian. Chapter 2 provides a new framework for developing new product formulas from existing ones. These ideas were developed in [8] and allow for one to combine deterministic and randomized product formulas into a single channel. This makes it particularly well suited to simulating environmental effects on a system.

One other property that our model for thermalization has is that it explicitly models the environment, keeping track of it's non-equilibrium state throughout the evolution. This allows for us to turn the problem of preparing the system in a thermal state to instead allow for our ancilla qubit to probe and extract information about the system. For example, if we know that the system is already in a thermal state we can allow it the thermalize our ancilla qubit. By measuring the temperature of the ancilla we can infer the temperature of the system, allowing us to do thermometry without creating an interaction model. This is advantageous for complicated quantum systems, as there currently only exists thermometry techniques for harmonic oscillators.

The remainder of this thesis is organized as follows. In Chapter 2 we introduce techniques for implementing existing product formulas on quantum computers, along with introducing new techniques for composing multiple simulation techniques for partitioned Hamiltonians. In Chapter 3 we demonstrate how these techniques can be used to prepare thermal quantum states on fault-tolerant quantum computers. These two chapters represent the main technical contributions of this thesis and draw on the papers [8] and [9] for the analytic ideas and [10] for numerical studies.

Chapter 2

Composite Simulations

The simulation of time-independent Hamiltonian dynamics is a fundamental primitive in quantum computing. To start, the computational problem of approximating the time dynamics of even k-local Hamiltonians (where k is a small constant) is BQP-Complete. This means that any computational problem that can be solved efficiently on a quantum computer can be efficiently reduced to a simulation problem.

The simulation of quantum systems remains one of the most compelling applications for future digital quantum computers [11–16]. As such, there are a plethora of algorithm options for compiling a unitary evolution operator $U(t) = e^{\{-iHt\}}$ to circuit gates [17–24]. Some of the simplest such algorithms are product formulas in which each term in a Hamiltonian $H = \sum_i h_i H_i$ is implemented as $e^{iH_i t}$. A product formula is then a particular sequence of these gates that approximates the overall operator $U(t)$. Two of the most well known product formula include Trotter-Suzuki Formulas [18,20,25,26] and the QDrift protocol in which terms are sampled randomly [24,27]. These two approaches are perhaps the most popular ancilla-free simulation methods yet discovered.

One of the main drawbacks of Trotter-Suzuki formulas is that each term in the Hamiltonian has to be included in the product formula regardless of the magnitude of the term. This leads to a circuit with a depth that scales at least linearly with the number of terms in H, typically denoted L. QDrift avoids this by randomly choosing which term to implement next in the product formula according to an importance sampling scheme in which higher weight terms have larger probabilities. The downside to QDrift is that it has the same asymptotic scaling with $\frac{t}{\varepsilon}$

as a first-order Trotter formula, meaning it is outperformed at large $\frac{t}{\varepsilon}$ by even a second-order Trotter formula.

In this paper we present a framework for combining simulation channels in a way that allows one to flexibly interpolate the gate cost tradeoffs between the individual channels. The primary example we study is the composition of Trotter-Suzuki and QDrift channels. This is motivated in some part as an effort to extend randomized compilers to include conditional probabilities and in some part to encapsulate progress in chemistry simulations of dropping small weight terms or shuffling terms around different time steps [28]. This latter concept was first developed with the idea of "coalescing" terms into "buckets" by Wecker et al. [28] and further explored by Poulin et al. [29]. They showed that grouping terms of similar sizes together to be skipped during certain Trotter steps led to negligible increases in error and reduced gate counts by about a factor of 10. Similar improvements are also seen in the randomized setting of [30]. In this work we extend on these ideas by placing a specific set of terms into a Trotter partition and the rest in a QDrift partition. This simple division can then be studied analytically and we are able to provide sufficient conditions on asymptotic improvements over completely Trotter or completely QDrift channels. Although we are not able to develop the idea of conditional samples in QDrift protocols, our procedure can be viewed as a specific subset of what a generic Markovian QDrift would look like. We briefly mention these generalizations in Section .

2.1 Related Work

Recent approaches have sought to use the advantages of randomized compilation as a subset of an overall simulation, such as the hybridized scheme for interaction picture simulations [31]. What separates these two works is that our approach offers a more flexible approach for generic time-independent simulation problems whereas the hybridized schemes are specifically tailored to taking advantage of the time dependence introduced by moving to an interaction picture. As such, the hybridized approach achieves asymptotic advantages when the size of the interaction picture term dominates the overall Hamiltonian. This typically occurs in instances in which the size of an operator is unbounded, which can occur in lattice field theory simulations or constrained systems. The way the hybridized scheme in [31] works is via a "vertical" stacking of simulation channels, for example one channel to handle the Interaction Picture rotations and then other channels on top of this to simulate the time-dependence it generates on the remaining

Hamiltonian terms. Our work instead remains in the Schrodinger time evolution picture and we perform a “horizontal” stacking of simulation techniques. By horizontal we mean for a given simulation time we split the Hamiltonian up into (potentially) disjoint partitions and simulate each partition for the full simulation time but with different techniques, such as Trotter or QDrift. These techniques allow us to achieve asymptotic improvements over either method for a loose set of assumptions.

There are two other simulation techniques that have been proposed recently that have a similar interpolation behavior between QDrift and Trotter channels. The first of these methods is the SparSto, or Stochastic Sparsification, technique by Ouyang, White, and Campbell [32]. The SparSto procedure randomly sparsifies the Hamiltonian and performs a randomly ordered first-order Trotter formula on the sampled Hamiltonian. They construct these probabilities such that the expected Hamiltonian is equal to the Hamiltonian being simulated. They then fix the expected number of oracle queries of the form $e^{iH_i t'}$ and give diamond distance bounds on the resulting channel error. The claim for interpolation between Trotter and QDrift is that one can fix the expected number of gates to be 1 for each time step, in which case the sparsification mimics QDrift, whereas if no sparsification is performed then the channel is simply implementing Trotter. They show that this allows for one to have reduced simulation error up to an order of magnitude on numerically studied systems as compared to Trotter or QDrift. One downside to these techniques is that the number of gates applied is a random variable, so making gate cost comparisons is rather difficult especially considering that no tail bounds on high gate cost sampled channels are provided. In [32] they prefer to fix the expected gate cost and analyze the resulting diamond norm error. In contrast, our procedures directly implement both QDrift and Trotter channels and have a fixed, deterministic gate cost.

The second method of note with both QDrift and Trotter behavior is that of Jin and Li [33]. They develop an analysis of the variance of a unitary consisting of a first-order Trotter sequence followed by a QDrift channel. They focus on bounding the Mean Squared Error (MSE) of the resulting channel and use a simple partition of the Hamiltonian terms based on spectral norm. Their partitioning scheme places all terms below some cutoff into the first-order Trotter sequence and all terms above the cutoff into the QDrift channel. Their main results show an interpolation of the MSE between 0 when the partitioning matches a solely Trotter channel and matching upper bounds for QDrift when all terms are randomly sampled. This work goes beyond the results from Jin and Li by providing an analysis of the diamond distance between an ideal evolution and our implemented channel, which is more useful analytically than the MSE, as well as providing upper bounds on the number of gates needed in an implementation to meet this diamond distance. In addition our work remains independent of specific partitioning schemes as much as possible and instead places restrictions on which partitions achieve improvements. In the interest of practicality we do show methods for partitioning that can be useful in both the first-order and higher-order Trotter cases. Specifically for higher-order Trotter formulas we give a probabilistic partitioning scheme that is easily computable and matches gate cost upper bounds in the extreme limits as our probabilities saturate the QDrift and Trotter limits.

The rest of the paper is organized as follows. We first provide a brief summary of the main results in Section 2.2. After reviewing known results and notation in Section 2.3, we explore methods for creating Composite channels using First-Order Trotter Formulas with QDrift in Section 2.4 as a warmup. This is broken down into three parts in which we find the gate cost for an arbitrary partition, we then give a method for producing a good partitioning, and then we analyze conditions in which a Composite channel can beat either first-order Trotter or QDrift channels. In Section 2.5 we then extend this framework to more general higher-order Trotter Formulas. This section mirrors the organization of the first-order Trotter section, namely we find the cost of an arbitrary partition, we give a method for producing a partition efficiently, and then we analyze when one could see improvements over the constituent channels. Finally, in Section 2.7 we discuss extensions to this model that allow a flexible interpolation between various types of product formulas that could be leveraged numerically.

2.2 Main Results

2.3 Preliminaries

In this section we will first introduce the necessary notation we will use and then state known results about Trotter-Suzuki formulas and QDrift channels. We work exclusively with time-independent Hamiltonians H in a 2^n dimensional Hilbert space $\mathcal{H}$. We also assume that H consists of L terms $H = \sum_{i=1}^{L} h_i H_i$ where h_i represents the spectral norm of the term, H_i is a Hermitian operator on $\mathcal{H}$, and $\|H_i\| = 1$. Note without loss of generality we can always assume $h_i \geq 0$, as we can always absorb the phase into the operator H_i itself. We use $\|M\|$ to refer to the spectral norm, or the magnitude of the largest singular value of M. We use λ to refer to the sum of h_i, namely $\lambda = \sum_i h_i$. We will also use subscripts on λ, such as λ_A to refer to sums of subsets of the terms of H. For example, if $H = 1H_1 + 2H_2 + 3H_3$ and $G = 1H_1 + 2H_2$, then $\lambda = 6$ and $\lambda_G = 3$.

We use $U(t)$ to refer to the unitary operator e^{iHt} and $\mathcal{U}(t)$ to refer to the channel $\rho \mapsto U(t)\rho U(t)^\dagger$. We will be particularly concerned with simulations of subsets of the terms of H, which we denote as follows. We typically work with a partition of H into two matrices $H = A + B$, and we let $A = \sum_i a_i A_i$ and $B = \sum_j b_j B_j$, where we have simply relabeled the relevant h_i and H_i into a's, b's, A's, and B's. This allows us to define the exact unitary time evolution operators $U_{A(t)} = e^{iAt}$ and channels $\mathcal{U}_{A(t)} = U_{A(t)}\rho U_{A(t)}^\dagger$, similarly defined for B.

Although much of the literature for Trotter-Suzuki formulas is written in terms of unitary operators $U = e^{iHt}$ acting on state vectors $|\psi\rangle$ for our purposes it will prove most natural to consider a product formula as a channel $\mathcal{U} = e^{iHt}\rho e^{-iHt}$ acting on a density matrix ρ. After reviewing known results on unitary constructions of Trotter-Suzuki formulas we give a straightforward extension of these bounds to channels.

2.3.1 Product Formulas

We now show how to implement basic product formulas, namely Trotter-Suzuki or just Trotter formulas as well as QDrift, assuming access to arbitrary single qubit unitaries and controlled NOT gates. We will first show how to implement an arbitrary Pauli rotation e^{iPt} for some Pauli string P using no additional ancilla qubits. Then we will define the Trotter-Suzuki construction and give heuristic evidence for the first order scaling. We avoid giving a rigorous proof and instead refer the reader to the canonical paper by Childs et. al [26]. Lastly, we will present the construction of QDrift by Campbell [24], providing a heuristic proof of correctness.

Definition 2.3.1.1 (Trotter-Suzuki Formulae): *Given a Hamiltonian H, let $S^{(1)}(t)$ denote the first-order Trotter-Suzuki time evolution operator and channel as*

$$\begin{aligned} S^{(1)}(t) &:= e^{ih_L H_L t} ... e^{ih_1 H_1 t} = \prod_{i=1}^{L} e^{ih_i H_i t}, \\ \mathcal{S}^{(1)}(\rho; t) &:= S^{(1)}(t) \cdot \rho \cdot S^{(1)}(t)^{\dagger}. \end{aligned} \tag{2.7}$$

This first order formula serves as the base case for the recursively defined higher-order formulas

$$\begin{aligned} S^{(2)}(t) &:= e^{ih_1 H_1 (\frac{t}{2})} ... e^{ih_L H_L (\frac{t}{2})} e^{ih_L H_L (\frac{t}{2})} ... e^{ih_1 H_1 (\frac{t}{2})} = S^{(1)}(t/2) \cdot S^{(1)}(-t/2)^{\dagger} \\ S^{(2k)}(t) &:= S^{(2k-2)}(u_k t) \cdot S^{(2k-2)}(u_k t) \cdot S^{(2k-2)}((1-4u_k)t) \cdot S^{(2k-2)}(u_k t) \cdot S^{(2k-2)}(u_k t) \\ \mathcal{S}^{(2k)}(\rho; t) &:= S^{(2k)}(t) \cdot \rho \cdot S^{(2k)}(t)^{\dagger}, \end{aligned} \tag{2.8}$$

where $u_k := \frac{1}{4 - 4^{\frac{1}{2k-1}}}$. In addition we define $\Upsilon_k := 2 \cdot 5^{k-1}$ as the number of "stages" in the higher-order product formula, although we will typically just write Υ when the order is apparent.

Despite their simplicity, Trotter-Suzuki formulas are fiendishly difficult to analyze. For decades the only error analysis that existed was worst-case analysis that often drastically overestimated the actual error. It was known that the first order expression depended on the commutator structure among the terms, but this was not generalized until 2021 in [26], 25 years after Lloyd's original work [34]. We will follow [26] and denote the expression that captures this commutator scaling as α_{comm}, and sometimes when space is needed this may be abbreviated to α_C when the context is clear, which we define as

$$\alpha_{\text{comm}}(H, 2k) := \sum_{\gamma_i \in \{1, ..., L\}} \left(\prod h_{\gamma_i} \right) \left\| \left[H_{\gamma_{2k+1}}, \left[H_{\gamma_{2k}}, \left[..., \left[H_{\gamma_2}, H_{\gamma_1} \right] ... \right] \right] \right] \right\|_{\infty}. \tag{2.9}$$

We will also make heavy use of the restriction of α_{comm} to subsets of the total Hamiltonian H. For example, if $H = A + B$ then we define the commutator structure over A as

$$\alpha_{\text{comm}}(A, 2k) := \sum_{\gamma_i \in \{1, ..., L_A\}} \left(\prod a_{\gamma_i} \right) \left\| \left[A_{\gamma_{2k+1}}, \left[A_{\gamma_{2k}}, \left[..., \left[A_{\gamma_2}, A_{\gamma_1} \right] ... \right] \right] \right] \right\|_{\infty}. \tag{2.10}$$

This then allows us to decompose the total commutator structure into 3 pieces: commutators that contain only terms from A, commutators that contain only terms from B, and commutators that contain *at least* one term from both A and B

$$\alpha_{\text{comm}}(H, 2k) = \alpha_{\text{comm}}(A, 2k) + \alpha_{\text{comm}}(B, 2k) + \alpha_{\text{comm}}(\{A, B\}, 2k). \tag{2.11}$$

This allows us to give the error associated with a a Trotter-Suzuki formula in the following theorem.

Theorem 2.3.1.1 (Trotter-Suzuki [26]): *Let $S^{(2k)}$ be the Trotter-Suzuki unitary as given in Definition 2.3.1.1 for the Hamiltonian $H = \sum_{i=1}^{L} h_i H_i$. Then the spectral norm of the difference between the Trotter-Suzuki formulas $S^{(1)}(t/r)$ and $S^{(2k)}(t/r)$ and the ideal evolution $U(t/r)$ is given by*

$$\begin{aligned} \left\| U(t/r) - S^{(1)}(t/r) \right\|_{\infty} &\leq \frac{t^2}{2r^2}\, \alpha_{\text{comm}}(H, 1), \\ \left\| U(t/r) - S^{(2k)}(t/r) \right\|_{\infty} &\leq \frac{(\Upsilon t)^{2k+1}}{r^{2k+1}(k + 1/2)} \alpha_{\text{comm}}(H, 2k). \end{aligned} \tag{2.12}$$

The associated operator exponential cost can be computed via standard time-slicing arguments as

$$\begin{aligned} C^{(1)}_{\text{Trot}}(H, t, \varepsilon) &= L \left\lceil \frac{t^2}{2\varepsilon} \sum_{i,j} \| \, [H_i, H_j] \, \| \right\rceil \\ C^{(2k)}_{\text{Trot}}(H, t, \varepsilon) &= \Upsilon L \left\lceil \frac{(\Upsilon t)^{1+1/2k}}{\varepsilon^{1/2k}} \left(4\alpha_{\text{comm}} \frac{(H, 2k)^{1/2k}}{2k + 1} \right) \right\rceil \end{aligned} \tag{2.13}$$

The complete proof of the above theorem is very nontrivial and beyond the scope of this thesis. See [26] for complete details, the proof of the higher order bounds can be found in Appendix E and the first order expression is found in Proposition 9 in Section V. Instead, we provide a heuristic proof for the first order error for completeness.

TODO: Probably should include a proof that spectral norm bounds on a unitary channel imply a diamond distance bound that is different only by a factor of 2.

TODO: Also probably should do a bit better proof below, can use a heuristic that all Taylor series of the exponential have the same special time t? or maybe just produce a bound?

Proof Heuristic First Order: Compute a Taylor Series for the Trotter formula and the ideal evolution. First the ideal evolution:

$$U(t/r) = e^{iH\frac{t}{r}} = \mathbb{1} + \frac{it}{r}H + O((t/r)^2). \tag{2.14}$$

Then the Trotter terms:

$$\prod_{i=1}^{L} e^{ih_i H_i t'} = \prod_{i=1}^{L}(\mathbb{1} + ih_i t' H_i + O(t'^2)) = \mathbb{1} + it' \sum_{i=1}^{L} h_i H_i + O(t'^2). \tag{2.15}$$

Pretty clear to see that in the difference $U(t/r) - S^{(1)}(t/r)$ the zeroth and first order terms vanish, leaving only the second order. □

2.3.2 Randomized Product Formulas

We now introduce QDrift [24], one of the first randomized compilers for quantum simulation. The main idea of QDrift is that instead of iterating through each term in the Hamiltonian to construct a product formula, or even a random ordering of terms as in [20], each exponential is chosen randomly from the list of terms in H. Each term is selected with probability proportional to it's spectral weight, the probability of choosing H_i is $\frac{h_i}{\sum_j h_j} =: \frac{h_i}{\|h\|}$, and then simulated for a time $\tau = \|h\|t$. This is the protocol for a single sample. As we will denote the portion of the Hamiltonian that we simulate with QDrift in later sections as B we let N_B denote the number of samples used.

Definition 2.3.2.1 (QDrift Channel): *Let N_B denote the number of samples, $\|h\| = \sum_{i=1}^{L} h_i$, and $\tau := \frac{\|h\|t}{N_B}$. The QDrift channel for a single sample is given as*

$$\mathcal{Q}(\rho; t, 1) := \sum_{i=1}^{L} \frac{h_i}{\|h\|} e^{-iH_i\|h\|t} \cdot \rho \cdot e^{+iH_i\|h\|t}, \tag{2.16}$$

and the QDrift channel for N_B samples is

$$\mathcal{Q}(t, N_B) := \mathcal{Q}(t/N_B, 1)^{\circ N_B}. \tag{2.17}$$

Once we have the channel defined we can then state the main results of [24].

Theorem 2.3.2.1 (QDrift Cost): *Given a Hamiltonian H, time t, and error bound ε, the ideal time evolution channel $\mathcal{U}(t)$ can be approximated using $N_B = \frac{4t^2\|h\|^2}{\varepsilon}$ samples of a QDrift channel. This approximation is given by the diamond distance*

$$\|\mathcal{U}(t) - \mathcal{Q}(t, N_B)\|_{\diamond} \leq \frac{4t^2\|h\|^2}{N_B}. \tag{2.18}$$

The number of operator exponentials N_B is then given as

$$C_{\text{QD}}(H, t, \varepsilon) = N_B = \left\lceil \frac{4t^2\|h\|^2}{\varepsilon} \right\rceil. \tag{2.19}$$

Proof Theorem 2.3.2.1: This one is also a Taylor's Series

$$\mathcal{U}_{\text{QD}}(t) \tag{2.20}$$

□

2.4 First Order Composite Channels

We now turn towards combining the two product formulas given in Section 2.3 in a Composite channel. We first will assume that the Hamiltonian has already been partitioned into two pieces $H = A + B$, where A will be simulated with a first order Trotter formula and B with QDrift. Given a fixed partitioning allows for us to compute the diamond distance error in the resulting channel, which then allows us to bound the number of operator exponentials needed to implement the channel. The resulting cost function will then be parametrized by the partitioning, which we can then use to determine an optimal partitioning algorithm. Finally, we give a specific instance in which a Composite channel can offer asymptotic improvements in query complexity over either a purely Trotter or QDrift channel.

2.4.1 Query Complexity

To analyze the error of our Composite channel we need to first reduce the overall time evolution channel $\rho \mapsto e^{-iHt}\rho e^{+iHt}$ into the simpler pieces that we can analyze with our Trotter and QDrift results. Assuming a partitioning $H = A + B$, where A consists of terms that we would like to simulate with Trotter and B has the terms we would like to sample from with QDrift. We now introduce the "outer-loop" error $E_{\{A,B\}}$ induced by this partitioning, which is as follows

$$E_{\{A,B\}}(t) := e^{-iHt}\rho e^{+iHt} - e^{-iBt}e^{-iAt}\rho e^{+iAt}e^{+iBt}. \tag{2.21}$$

We use the phrase "outer-loop" as this decomposition is done before any simulation channels are implemented. This gives the first order Composite channel $\mathcal{C}$ as

$$\mathcal{C}^{(1)}(t) := \mathcal{Q}_B(t, N_B) \circ \mathcal{S}_A^{(1)}(t). \tag{2.22}$$

We will first bound the error of this approximation to the ideal evolution. This error bound will then allow us to bound the number of exponentials needed to approximate the ideal dynamics.

Lemma 2.4.1.1 (First-Order Composite Error): *Given a Hamiltonian H partitioned into a first order Trotter term A and QDrift term B such that $H = A + B$, the first order Composite Channel $\mathcal{C}^{(1)}$ has an error of at most*

$$\left\|\mathcal{U}(t) - \mathcal{C}^{(1)}(t)\right\|_\diamond \leq t^2\left(\sum_{i,j} a_i a_j \left\|\left[A_i, A_j\right]\right\| + \sum_{i,j}\left\|\left[A_i, B_j\right]\right\| + \frac{4\|h_B\|^2}{N_B}\right). \tag{2.23}$$

Proof: We will first use an outer-loop decomposition to get the error associated by our partitioning, note we temporarily supress arguments of (t) for clarity,

$$\begin{aligned}\left\|\mathcal{U} - \mathcal{C}^{(1)}\right\|_\diamond &= \left\|\mathcal{U} - \mathcal{U}_B \circ \mathcal{U}_A + \mathcal{U}_B \circ \mathcal{U}_A - \mathcal{C}^{(1)}\right\| \\ &\leq \left\|\mathcal{U} - \mathcal{U}_B \circ \mathcal{U}_A\right\|_\diamond + \left\|\mathcal{U}_B \circ \mathcal{U}_A - \mathcal{C}^{(1)}\right\|_\diamond.\end{aligned} \tag{2.24}$$

We then can bound the leftmost term using the error decomposition

$$\left\|\mathcal{U} - \mathcal{U}_B \circ \mathcal{U}_A\right\|_\diamond \leq 2\left\|U - U_B \cdot U_A\right\|_\infty \leq t^2 \sum_{i,j} a_i b_j \left\|\left[A_i, B_j\right]\right\|_\infty. \tag{2.25}$$

And the rightmost term can be bounded using the subadditivity of the diamond distance

$$\begin{aligned}\left\|\mathcal{U}_B \circ \mathcal{U}_A - \mathcal{C}^{(1)}\right\|_\diamond &= \left\|\mathcal{Q}_B \circ \mathcal{S}_A^{(1)} - \mathcal{U}_B \circ \mathcal{U}_A\right\|_\diamond \\ &\leq \left\|\mathcal{U}_B - \mathcal{Q}_B\right\|_\diamond + \left\|\mathcal{U}_A - \mathcal{S}_A^{(1)}\right\|_\diamond \\ &\leq \frac{4\|h_B\|t^2}{N_B} + t^2\sum_{i,j} a_i a_j \left\|\left[A_i, A_j\right]\right\|_\infty.\end{aligned} \tag{2.26}$$

Substituting Equation (2.25) and Equation (2.26) into Equation (2.24) yields the statement. □

Now that we have an upper bound on first order error for an arbitrary t we can leverage this into a bound on the number of operator exponentials to reach arbitrary error ε using standard time-slicing arguments. By letting $t \to t/r$ and then repeating our Composite channel r times we can control the accumulated error from each step. One of the beautiful features of product formulas is that this time-slicing leads to an overall reduction in the error, or in other words

$$\lim_{r\to\infty} \left(e^{iA\frac{t}{r}} e^{iB\frac{t}{r}}\right)^r = e^{i(A+B)t}. \tag{2.27}$$

The following theorem utilizes a quantitative variant of the above, along with the error bounds we just proved, to provide the first order Composite cost Theorem.

Theorem 2.4.1.1 (First-Order Composite Cost): *Given a time t, error bound ε, and a partitioned Hamiltonian $H = A + B$, the first order Composite Channel $\mathcal{C}^{(1)}$ approximates the ideal time evolution operator $\left\| \mathcal{U}(t) - \mathcal{C}^{(1)}(t/r)^{\circ r} \right\|_{\diamond} \leq \varepsilon$ using no more than*

$$C^{(1)}_{\text{Comp}} = (L_A + N_B) \left\lceil \frac{t^2}{\varepsilon} \left(\sum_{i,j} a_i a_j \| [A_i, A_j] \| + \sum_{i,j} \| [A_i, B_j] \| + \frac{4\|h_B\|^2}{N_B} \right) \right\rceil \tag{2.28}$$

operator exponential queries.

Proof: We first will use the fact that since H commutes with itself the time evolution operator can be decomposed into r steps as $\mathcal{U}(t) = \mathcal{U}(t/r)^{\circ r}$. Then we can use the sub-additivity of the diamond norm with respect to channel composition to get the bound

$$\left\| \mathcal{U}(t) - \mathcal{C}^{(1)}(t/r)^{\circ r} \right\|_{\diamond} = \left\| \mathcal{U}(t/r)^{\circ r} - \mathcal{C}^{(1)}(t/r)^{\circ r} \right\|_{\diamond} \leq r \left\| \mathcal{U}(t/r) - \mathcal{C}^{(1)}(t/r) \right\|_{\diamond}. \tag{2.29}$$

Now we utilize Lemma 2.4.1.1 to upper bound the single step error

$$\left\| \mathcal{U}(t) - \mathcal{C}^{(1)}(t/r)^{\circ r} \right\|_{\diamond} \leq r \left(\frac{t^2}{r^2} \right) \left(\sum_{i,j} a_i a_j \| [A_i, A_j] \| + \sum_{i,j} \| [A_i, B_j] \| + \frac{4\|h_B\|^2}{N_B} \right). \tag{2.30}$$

In order for the above to be upper bounded by ε we require

$$r \geq \left(\frac{t^2}{\varepsilon} \right) \left(\sum_{i,j} a_i a_j \| [A_i, A_j] \| + \sum_{i,j} \| [A_i, B_j] \| + \frac{4\|h_B\|^2}{N_B} \right), \tag{2.31}$$

and since increasing r only increases the number of operator exponentials used we simply set r to be the ceiling of the right hand side. This then yields the theorem as we have r applications of $\mathcal{C}^{(1)}$ and each application uses L_A operator exponentials for the Trotter channel and N_B samples of the QDrift channel. □

2.4.2 First-Order Parameter Settings

Our next task will be to determine "parameter settings" that optimize this gate cost, namely the partitioning $A+B$ and setting the number of QDrift samples N_B. To do this it will prove useful to have a continuous, non-integer variant of the gate cost expression which we define as

$$\tilde{C}_{\text{Comp}}^{(1)} := (L_A + N_B)\frac{t^2}{\varepsilon}\left(\sum_{i,j} a_i a_j \| [A_i, A_j] \| + \sum_{i,j} \| [A_i, B_j] \| + \frac{4\|h_B\|^2}{N_B}\right). \tag{2.32}$$

This is the same as $C_{\text{Comp}}^{(1)}$ but without the ceiling operation $\lceil \cdot \rceil$. Although a user could use any value of N_B they want, such as always setting $N_B = 1$, we provide the following setting for N_B that is optimal with respect to the continuous variant of the gate cost.

Lemma 2.4.2.1: *Let $\tilde{C}_{\text{Comp}}^{(1)}$ denote the continuous relaxtion to the cost of a first-order Composite channel with a given partitioning $H = A + B$. The optimal assignment of the number of QDrift samples N_B with respect to $\tilde{C}_{\text{Comp}}^{(1)}$ is given by*

$$N_B = \frac{2\|h_B\|\sqrt{L_A}}{\sqrt{\left(\sum_{i,j} a_i a_j \| [A_i, A_j] \| + \sum_{i,j} a_i b_j \| [A_i, B_j] \|\right)}}. \tag{2.33}$$

This assignment is not valid if both $[A_i, A_j] = 0$ for all A_i, A_j and $[A_i, B_j] = 0$ for all A_i and B_j.

Proof: We first compute the derivative of $\tilde{C}_{\text{Comp}}^{(1)}$ with respect to N_B as

$$\frac{\partial \tilde{C}_{\text{Comp}}}{\partial N_B} = \frac{t^2}{\varepsilon}\left[\sum_{i,j} a_i a_j \| [A_i, A_j] \| + \sum_{i,j} a_i b_j \| [A_i, B_j] \| - \frac{4\|h_B\|^2 L_A}{N_B^2}\right]. \tag{2.34}$$

Setting this equal to 0 and solving for N_B yields Equation (2.33). We then compute the second derivative as

$$\frac{\partial^2 \tilde{C}_{\text{Comp}}^{(1)}}{\partial N_B^2} = \frac{4t^2\|h_B\|^2 L_A}{\varepsilon N_B^3}, \tag{2.35}$$

which is always positive and therefore indicates that the optima found is a minimal cost with respect to N_B. □

There is still one remaining problem with the first-order Composite channel that we must address before we can compare it to existing product formulas: partitioning. Up until now we have assumed that a partitioning was given, but this is not a realistic assumption to make. There are many heuristics that one could, and most likely should, take advantage of when implementing an actual channel. For example, in a chemistry simulation one can put the nuclei-electron interactions, which are typically stronger than the electron-electron interactions, into Trotter and use a spectral norm cutoff to determine the remainder. On could construct a Hubbard like model on a grid but with long-range interactions, treating the "tunneling" kinetic term and nearest neighbor interactions with Trotter and then sampling the long-range interactions with QDrift. These kinds of heuristics will most likely be important in reducing simulation costs but will ultimately be application dependent.

We would like to provide a general purpose algorithm, one that works for any Hamiltonian and matches our intuition that the above heuristics capture. The algorithm we propose is a gradient based scheme that is based on a weighting of the Hamiltonian terms H_i. This weighting is based on the following trick, for every term H_i we introduce a parameter w_i that accounts for the weight of H_i in the Trotter partition. Then our Hamiltonian can be written as

$$H = \sum_i h_i H_i = \sum_i (w_i h_i H_i + (1 - w_i) h_i H_I) = \sum_i w_i h_i H_i + \sum_i (1 - w_i) h_i H_i. \tag{2.36}$$

Then our partitioning is just $A = \sum_i w_i h_i H_i$ and $B = \sum_i (1 - w_i) h_i H_i$. We also would like to keep our interpretation of $w_i h_i$ and $(1 - w_i) h_i$ as spectral norms of the respective term in A and B, so we restrict $w_i \in [0, 1]$. In this sense the weights could be thought of as probabilities, but we will not make use of any expectations or other probabilistic notions here so they should just be thought of as weight parameters. We will make use of a probabilistic variant of this scheme in Section 2.5.

Once an initial weighting of each term is chosen, say $w_i = 1$ for the term with the largest spectral norm $h_{\max}$ and $w_i = 0$ for every other term or maybe $w_i = \frac{1}{2}$ for all i, we propose a greedy algorithm for computing a new set of weights ${w_i}'$. This greedy algorithm is based on the following gradient calculation of the weighted first order Composite channel cost

$$\frac{\partial \tilde{C}^{(1)}_{\text{Comp}}}{\partial w_m} = (L_A + N_B) \frac{t^2}{\varepsilon} \left(h_m \sum_j h_j \| [H_j, H_m] \| - \frac{8 h_m \sum_i (1 - w_i) h_i}{N_B} \right). \tag{2.37}$$

This gradient could be used in gradient descent, once the derivatives for all w_m are computed and put into a vector one could update the parameters with ${w_i}' = w_i - \eta \nabla_w \tilde{C}^{(1)}_{\text{Comp}}$, where η is some learning rate.

Although this gradient descent based algorithm would be relatively easy to compute and implement in practice, it does not take advantage of the fact that our gradients are not only analytic, but linear with respect to a *single* parameter w_m. This means that if we only update a single weight parameter at a time we can find an optimal assignment of w_m with respect to the partial derivative given above. Setting the derivative equal to 0 and solving for w_m yields

$$\frac{\partial \tilde{C}^{(1)}_{\text{Comp}}}{\partial w_m} = 0 \Longrightarrow {w_m}' = 1 - \sum_{i \neq m} \frac{h_i}{h_m} \left(\frac{\| [H_i, H_m] \|}{8} - (1 - w_i) \right). \tag{2.38}$$

Once ${w_m}'$ is determined then we can move on to ${w_{m+1}}'$ until all parameters have been updated, at which point we can repeat until the parameters do not change. We unfortunately cannot provide more analysis, such as how many iterations this process will take, due to the coupling of the parameters.

Although our expression does not give exact partitionings that can be calculated in one go, our greedy algorithm based on Equation (2.38) does capture two key pieces of intuition that we suspect good partitions will have. The first is that if a term H_m commutes with every other term ($\forall i \neq m$ $[H_i, H_m] = 0$) then we have that ${w_m}' = 1 + \sum_{i \neq m} \frac{h_i(1-w_i)}{h_m} \geq 1$, so we then restrict the update to ${w_m}' = 1$. This implies that H_m is completely placed into the Trotter partition, as we would expect. The other piece of intuition is that smaller terms are pushed more towards the QDrift side of the partitioning. This can be seen from Equation (2.38) while considering the limit as $h_m \to 0$. If we assume that $\| [H_i, H_m] \| \geq (1 - w_i)$ on average, then the expression becomes $w_m \to -\infty$ in this limit, which we stop at 0. In total, this indicates that large terms that do not commute with small terms with most of their weight in the Trotter partition tend to push the weight of small terms more towards QDrift.

2.4.3 Comparison with Trotter and QDrift

Now that we have analyzed the cost and given a partitioning scheme for the first order Composite channel, we would like to know under what conditions this Composite channel can lead to comparable errors with lower gate cost. Instead of aiming to show that a Composite channel will outperform either first-order Trotter or QDrift for arbitrary Hamiltonians, we instead illustrate a concrete setting in which we achieve guaranteed asymptotic improvements. In later sections we are able to show more generic conditions in which asymptotic improvements can be obtained for higher-order formulas.

To start, let H be a Hamiltonian that has a partitioning into A and B such that the following conditions hold.

1. The number of non-zero commutators between terms in A scales with the square root of L_A. Mathematically,
$$\left|\left\{(i,j) : \left\| [A_i, A_j] \right\| \neq 0\right\}\right| =: N_{\text{nz}}^2 \in o(L_A). \tag{2.39}$$
2. The strength of the B terms, $\|h_B\| = \sum_i b_i$, is asymtotically less than the maximum commutator norm divided by the number of terms in A
$$\|h_B\| \leq \frac{a_{\max} N_{\text{nz}}^2}{L_A}, \tag{2.40}$$
where $a_{\max} = \max_i a_i$.
3. The number of terms in the A partition is vanishingly small compared to the total number of terms: $L_A \in o(L)$.

Next, we can use the optimal N_B value from Lemma 2.4.2.1 and Equation (2.40) to show that
$$N_B^{-1} \in O\left(\frac{1}{\|h_B\|}\sqrt{N_{\text{nz}}^2 a_{\max}^2 + L_A a_{\max} \|h_B\|}\right) = O\left(\frac{N_{\text{nz}} a_{\max}}{\|h_B\|}\right). \tag{2.41}$$
Similarly we have $N_B \in O\left(\frac{\|h_B\|\sqrt{L_A}}{a_{\max} N_{\text{nz}}}\right)$. Thus, Theorem 2.4.1.1 gives us the asymptotic number of operator exponentials as
$$\begin{aligned} C_{\text{Comp}}^{(1)} &\in O\left(\frac{t^2}{\varepsilon}\left(L_A + \frac{\|h_B\|\sqrt{L_A}}{a_{\max} N_{\text{nz}}}\right)\left(a_{\max}^2 N_{\text{nz}}^2 + L_A a_{\max}\|h_B\| + \frac{\|h_B\| N_{\text{nz}} a_{\max}}{\sqrt{L_A}}\right)\right) \\ &\in O\left(\frac{t^2 L_A}{\varepsilon}(a_{\max}^2 N_{\text{nz}}^2)\right) \\ &\in o\left(\frac{t^2}{\varepsilon} L_A^2 a_{\max}^2\right). \end{aligned} \tag{2.42}$$
The lowest order Trotter formula for this simulation has the following asymptotic operator exponential cost, as given in
$$C_{\text{Trot}}^{(1)} \in O\left(\frac{t^2}{\varepsilon}(L N_{\text{nz}}^2 a_{\max}^2)\right) \in o\left(\frac{t^2}{\varepsilon} L L_A a_{\max}^2\right) \subseteq \omega\left(C_{\text{Comp}}^{(1)}\right). \tag{2.43}$$
For QDrift we can use Theorem 2.3.2.1 to compute

$$
\begin{aligned}
C_{\mathrm{QD}} &\in O\left(\frac{t^2}{\varepsilon}(L_A a_{\max} + \|h_B\|)^2\right) \\
&\in O\left(\frac{t^2 L_A^2 a_{\max}^2}{\varepsilon}\left(1 + \left(\frac{N_{\mathrm{nz}}}{L_A}\right)^2\right)\right) \\
&\in O\left(\frac{t^2 L_A^2 a_{\max}^2}{\varepsilon}\right) \\
&\in \omega\left(C_{\mathrm{Comp}}^{(1)}\right). \qquad (2.44)
\end{aligned}
$$

This shows that the Composite channel has asymptotically better cost over the methods it composes, i.e. $C_{\mathrm{Comp}}^{(1)} \in o\left(\min\left(C_{\mathrm{Trot}}^{(1)}, C_{\mathrm{QD}}\right)\right)$.

Although this example may be a little contrived, it does show in a completely rigorous manner that there do exist scenarios in which even first order Composite techniques could provide significant constant factor improvements. This provides strong evidence that more detailed research is needed to understand when Composite techniques can provide advantages. We provide further numeric evidence comparing Composite techniques to Trotter and QDrift for some small systems in Section 2.6.

2.5 Higher Order Composite Channels

We now move on from first-order Trotter formulas to arbitrary higher-order Trotter formulas. To analyze this case there are a few distinct differences with the first-order channels. The first is that we now have a choice for what order formula we would like to use for the outer-loop decomposition. Previously, for the first-order decomposition we used $\mathcal{U}_B \circ \mathcal{U}_A$, but it will prove useful in our analysis to match the outer-loop order with the inner-loop Trotter order. For example, a second order outer-loop decomposition would look like $\mathcal{U}_A(t/2) \circ \mathcal{U}_B(t) \circ \mathcal{U}_A(t/2)$, where we combined the two innermost $\mathcal{U}_B(t/2)$ for compactness. The next difference is that the time scaling between QDrift, Trotter, and the outer-loop errors could all be of different orders in $\frac{t}{r}$ which leads to a non-analytically solvable polynomial in r. The last issue that we address is that the commutator structure is no longer quadratic with respect to the Hamiltonian spectral norms, so we cannot follow the term weighting partitioning scheme from the first-order case. We will follow the same organizational structure as the first-order case and first set up our definitions and bound the diamond distance error, then compute the number of $e^{iH_i t}$ queries, followed by developing a partitioning scheme, and finally discuss the cost comparisons between our Composite channel and its constituents.

2.5.1 Query Complexity

In order to determine the number of queries needed for a Composite channel to approximate $\mathcal{U}$ we first need to bound the diamond distance error for a single iteration. We will then use time-slicing arguments similar to the proof of Theorem 2.4.1.1 to compute the number of operator exponentials required for an accurate approximation. First, we need to give a rigorous definition of the higher order Composite channel.

Definition 2.5.1.1 (Higher Order Composite Channel): *Given a Hamiltonian H partitioned into two terms A and B, let $\mathcal{C}^{(2k,2l)}$ denote the associated Composite channel that utilizes a $2k^{\text{th}}$ order inner-loop for the Trotter-Suzuki partition $\mathcal{S}_A^{(2k)}$ and has a $2l^{\text{th}}$ order outer-loop. The outer-loop construction for the Composite channel can be constructed recursively from the base case for $l = 1$, which is given by*

$$\mathcal{C}^{(2k,2)}(t) := \mathcal{Q}_B(t/2) \circ \mathcal{S}_A^{(2k)}(-t/2)^\dagger \circ \mathcal{S}_A^{(2k)}(t/2) \circ \mathcal{Q}_B(t/2), \tag{2.45}$$

and the higher-order outer-loop Composite channels are recursively defined as

$$\mathcal{C}^{(2k,2l)}(t) := \mathcal{C}^{(2k,2l-2)}(u_l t)^{\circ 2} \circ \mathcal{C}^{(2k,2l-2)}((1-4u_l)t) \circ \mathcal{C}^{(2k,2l-2)}(u_l t)^{\circ 2}, \tag{2.46}$$

where u_l and the number of stages Υ are the same as in Definition 2.3.1.1. We will typically ignore the distinction between inner and outer loops and use $\mathcal{C}^{(2k)} = \mathcal{C}^{(2k,2k)}$.

Lemma 2.5.1.1 (Higher-Order Composite Error): *The diamond distance of a single higher-order Composite channel to the ideal time evolution channel is upper bounded as*

$$\left\|\mathcal{U}(t) - \mathcal{C}^{(2k)}(t)\right\|_\diamond \leq 2\frac{(\Upsilon t)^{2k+1}}{k+1/2}(\alpha_C(\{A,B\},2k) + \Upsilon\alpha_C(A,2k)) + \Upsilon\frac{4\|h_B\|^2 t^2}{N_B}. \tag{2.47}$$

We will use the following definitions for brevity

$$P(t) := t^{2k+1}\frac{2\Upsilon^{2k+1}}{k+1/2}(\alpha_{\text{comm}}(\{A,B\},2k) + \alpha_{\text{comm}}(A,2k))$$

$$Q(t) := t^2\frac{4\Upsilon\|h_B\|^2}{N_B}. \tag{2.48}$$

Proof of Lemma 2.5.1.1:

$$\begin{aligned}\left\|\mathcal{U}(t)-\mathcal{C}^{(2k)}(t)\right\|_\diamond &\leq \left\|\mathcal{U}(t)-\mathcal{S}^{(2k)}(\{A,B\},t)\right\|_\diamond + \left\|\mathcal{S}^{(2k)}(\{A,B\},t)-\mathcal{C}^{(2k)}(t)\right\|_\diamond \\ &\leq 2\left\|e^{iHt}-S^{(2k)}(\{A,B\},t)\right\| + \left\|\mathcal{S}^{(2k)}(\{A,B\},t)-\mathcal{C}^{(2k)}(t)\right\|_\diamond . \end{aligned} \quad (2.49)$$

We can use Theorem 2.3.1.1 to bound the outer-loop error on the left as

$$\left\|e^{iHt}-S^{(2k)}(\{A,B\},t)\right\| \leq \frac{(\Upsilon t)^{2k+1}}{k+1/2}\alpha_{\text{comm}}(\{A,B\},2k). \quad (2.50)$$

We then use an inductive proof to argue that the inner-loop errors can be bounded as

$$\left\|\mathcal{S}^{(2l)}(\{A,B\},t)-\mathcal{C}^{(2k,2l)}(t)\right\|_\diamond \leq \Upsilon\left(\left\|\mathcal{U}_A(t)-\mathcal{S}_A^{(2k)}(t)\right\|_\diamond + \left\|\mathcal{U}_B(t)-\mathcal{Q}_B(t)\right\|_\diamond\right), \quad (2.51)$$

where the induction is over the outer-loop indexing of $2l$.

- **Base Case ($2l=1$):**

$$\begin{aligned}&\left\|\mathcal{C}^{(2k,2)}(t)-\mathcal{S}^{(2)}(\{A,B\})(t)\right\|_\diamond \\ &= \left\|\mathcal{Q}_B(t/2)\circ\mathcal{S}_A^{(2k)}(-t/2)^\dagger\circ\mathcal{S}_A^{(2k)}(t/2)\circ\mathcal{Q}_B(t/2)-\mathcal{U}_B(t/2)\circ\mathcal{U}_A(-t/2)^\dagger\circ\mathcal{U}_A(t/2)\circ\mathcal{U}_B(t/2)\right\|_\diamond \\ &\leq 2\left\|\mathcal{U}_A(t/2)-S_A^{(2)}(t/2)\right\|_\diamond + 2\left\|\mathcal{U}_B(t/2)-\mathcal{Q}_B(t/2)\right\|_\diamond .\end{aligned} \quad (2.52)$$

  Since $\Upsilon_1=2$ this matches the induction hypothesis.

- **Inductive Step:** In this scenario we assume that the hypothesis in Equation (2.51) holds for $2l-2$ and we would like to show it holds for $2l$. We do so via the recursive structure given in Equation (2.46) and Definition 2.3.1.1, which allows us to express the hypothesis as

$$
\begin{aligned}
&\left\|\mathcal{C}^{(2k,2l)}(t)-\mathcal{S}^{(2l)}(\{A,B\},t)\right\|_{\diamond} \\
&=\Big\|\mathcal{C}^{(2k,2l-2)}(u_l t)^{\circ 2}\circ\mathcal{C}^{(2k,2l-2)}((1-4u_l)t)\circ\mathcal{C}^{(2k,2l-2)}(u_l t)^{\circ 2} \\
&\quad -\mathcal{S}^{(2l-2)}(\{A,B\},u_l t)^{\circ 2}\circ\mathcal{S}^{(2l-2)}(\{A,B\},(1-4u_l)t)\circ\mathcal{S}^{(2l-2)}(\{A,B\},u_l t)^{\circ 2}\Big\|_{\diamond} \\
&\leq 4\left\|\mathcal{C}^{(2k,2l-2)}(u_l t)-\mathcal{S}^{(2l-2)}(\{A,B\},u_l t)\right\|_{\diamond} \\
&\quad +\left\|\mathcal{C}^{(2k,2l-2)}((1-4u_l)t)-\mathcal{S}^{(2l-2)}(\{A,B\},(1-4u_l)t)\right\|_{\diamond} \\
&\leq 4\Upsilon_{l-1}\left(\left\|\mathcal{U}_A(t)-\mathcal{S}_A^{(2k)}(t)\right\|_{\diamond}+\left\|\mathcal{U}_B(t)-\mathcal{Q}_B(t)\right\|_{\diamond}\right) \\
&\quad +\Upsilon_{l-1}\left(\left\|\mathcal{U}_A(t)-\mathcal{S}_A^{(2k)}(t)\right\|_{\diamond}+\left\|\mathcal{U}_B(t)-\mathcal{Q}_B(t)\right\|_{\diamond}\right) \\
&\quad =5\Upsilon_{l-1}\left(\left\|\mathcal{U}_A(t)-\mathcal{S}_A^{(2k)}(t)\right\|_{\diamond}+\left\|\mathcal{U}_B(t)-\mathcal{Q}_B(t)\right\|_{\diamond}\right) \\
&=\Upsilon_{l}\left(\left\|\mathcal{U}_A(t)-\mathcal{S}_A^{(2k)}(t)\right\|_{\diamond}+\left\|\mathcal{U}_B(t)-\mathcal{Q}_B(t)\right\|_{\diamond}\right).
\end{aligned}
\tag{2.53}
$$

Therefore the inductive step holds.

One point of emphasis we would like to make is that we are explicitly not utilizing the smaller times that come with the recursive outer-loop step. This is simply due to the difficulty of bookkeeping for each different time step used and it will be sufficient to use the upper bound of t. We can now continue with our proof of the lemma by substituting in the known Trotter-Suzuki and QDrift error terms, leading us to

$$
\begin{aligned}
\left\|\mathcal{U}(t)-\mathcal{C}^{(2k)}(t)\right\|_{\diamond} &\leq 2\frac{(\Upsilon t)^{2k+1}}{k+1/2}\alpha_{\text{comm}}(\{A,B\},2k)+\Upsilon\left(\left\|\mathcal{U}_A(t)-\mathcal{S}_A^{(2k)}(t)\right\|_{\diamond}+\left\|\mathcal{U}_B-\mathcal{Q}_B(t)\right\|_{\diamond}\right) \\
&\leq 2\frac{(\Upsilon t)^{2k+1}}{k+1/2}(\alpha_{\text{comm}}(\{A,B\},2k)+\Upsilon\alpha_{\text{comm}}(A,2k))+\Upsilon\frac{4\|h_B\|^2t^2}{N_B}.
\end{aligned}
\tag{2.54}
$$

Note that the extra factor of Υ in front of $\alpha_{\text{comm}}(A)$ comes from the fact that we have Υ copies of the A simulation channel as opposed to just one outer-loop decomposition. □

Now that we have bounded the diamond distance error for a single time step we can proceed with our time-slicing arguments to produce a controllable error bound. This will lead us to our final expression for the query cost of a higher order composite method.

Theorem 2.5.1.1 (Higher-Order Composite Cost): *Given a time t, error bound ε, and a partitioned Hamiltonian $H = A + B$ the $2k^{\text{th}}$ order Composite channel $\mathcal{C}^{(2k)}$ utilizes at most*

$$C^{(2k)}_{\text{Comp}} \le \Upsilon(\Upsilon L_A + N_B)\left\lceil \frac{4^{\frac{1}{2k}}(\Upsilon t)^{1+\frac{1}{2}k}}{((2k+1)\varepsilon)^{\frac{1}{2k}}}(\alpha_C(\{A,B\}) + \Upsilon\alpha_C(A)) + \frac{4\Upsilon\|h_B\|^2 t^2}{N_B\varepsilon}\right\rceil, \tag{2.55}$$

gates,where the α_C are both of order $2k$, to meet the error budget given by

$$\left\|\mathcal{U}(t) - \mathcal{C}^{(2k)}(t/r)^{\circ r}\right\|_{\diamond} \le \varepsilon. \tag{2.56}$$

Using the upper bounds provided for Trotter-Suzuki and QDrift evolution channels and defining

$$q_B := \frac{\alpha_{\text{Comm}}(B, 2k)}{\alpha_{\text{Comm}}(H, 2k)} \tag{2.57}$$

to capture the amount of "commutator structure" of H that is contained in B, we can rewrite this upper bound as

$$C^{(2k)}_{\text{Comp}} \le \Upsilon(\Upsilon L_A + N_B)\left\lceil \frac{C^{(2k)}_{\text{Trot}}(H,t,\varepsilon)}{\Upsilon^{1-1/2k}}\frac{(1-q_B)^{1/2k}}{L} + C_{\text{QD}}(H,t,\varepsilon)\frac{\Upsilon}{N_B}\left(\frac{\|h_B\|}{\|h\|}\right)^2\right\rceil. \tag{2.58}$$

Proof: We start off by utilizing standard time-slicing arguments to bound our single step distance

$$\left\|\mathcal{U}(t) - \mathcal{C}^{(2k)}(t/r)^{\circ r}\right\|_{\diamond} = \left\|\mathcal{U}(t/r)^{\circ r} - \mathcal{C}^{(2k)}(t/r)^{\circ r}\right\|_{\diamond} \le r\left\|\mathcal{U}(t/r) - \mathcal{C}^{(2k)}(t/r)\right\|_{\diamond}. \tag{2.59}$$

Using our results in Lemma 2.5.1.1 we can then bound the single time step error as

$$\left\|\mathcal{U}(t/r) - \mathcal{C}^{(2k)}(t/r)\right\|_{\diamond} \le \frac{P(t)}{r^{2k+1}} + \frac{Q(t)}{r^2}. \tag{2.60}$$

Our goal is to find a lower bound on r that will guarantee that the above is less than ε. In previous arguments we had monomials in r which allowed us to take roots to compute a bound, but the polynomial $ar^n + br^2 = c$ does not have closed form solutions for $n > 5$. We could try to provide closed solutions for second and fourth order Trotter-Suzuki formulas but we will instead provide a generic bound that will work for all higher order expressions.

Our route to constructing such abound comes from requiring the following inequalities be satisfied for $r_{\min} < r$

$$\frac{P(t)}{r^{2k+1}} + \frac{Q(t)}{r^2} \le \frac{P(t)}{r_{\min}^{2k+1}} + \frac{Q(t)}{r_{\min}^2} \le \frac{\varepsilon}{r} \le \frac{\varepsilon}{r_{\min}}. \tag{2.61}$$

We can then create the intermediate inquality

$$\frac{P(t)}{r^{2k+1}} + \frac{Q(t)}{r^2} \leq \frac{P(t)}{r^2 r_{\min}^{2k-1}} + \frac{Q(t)}{r^2} \leq \frac{P(t)}{r_{\min}^{2k+1}} + \frac{Q(t)}{r_{\min}^2} \leq \frac{\varepsilon}{r}. \tag{2.62}$$

Pulling these powers of r out allow us to simplify the inequality to

$$\frac{1}{\varepsilon}\left(\frac{P(t)}{r_{\min}^{2k-1}} + Q(t)\right) \leq r. \tag{2.63}$$

Our final inequality then comes from using only powers of $r_{\min}$ and noting the fact that $Q(t) > 0$ for all t. We have

$$\begin{aligned} \frac{P(t)}{r_{\min}^{2k+1}} &< \frac{P(t)}{r_{\min}^{2k+1}} + \frac{Q(t)}{r_{\min}^2} \leq \frac{\varepsilon}{r_{\min}} \\ \frac{P(t)}{\varepsilon} &< r_{\min}^{2k} \\ \left(\frac{P(t)}{\varepsilon}\right)^{1/2k} &< r_{\min}, \end{aligned} \tag{2.64}$$

therefore achieving our bound on r, which can be thought of as simply taking r large enough to ensure the Trotterized error is sufficiently small.

By plugging Equation (2.64) into Equation (2.63) yields an explicit lower bound on r as

$$\left(\frac{P(t)}{\varepsilon}\right)^{1/2k} + \frac{Q(t)}{\varepsilon} < r. \tag{2.65}$$

This matches the intuition developed from Trotter-Suzuki formulas in which the error decreases rapidly with the order of the formula, but leads to overall higher gate counts due to exponentially increasing constant factors, namely Υ_k. We now can write down the number of operator exponentials explicitly. As we have Υ stages of interleaved product formulas and each stage has one application of a $2k$ order Trotter-Suzuki formula and one QDrift channel with N_B samples we have $\Upsilon(\Upsilon L_A + N_B)$ operator exponentials per time-slice. By taking the ceiling of the derived bound on r we arrive at

$$C_{\text{Comp}}^{(2k)} \leq \Upsilon(\Upsilon L_A + N_B)\left\lceil \left(\frac{P(t)}{\varepsilon}\right)^{1/2k} + \frac{Q(t)}{\varepsilon} \right\rceil, \tag{2.66}$$

plugging in the definitions of $P(t)$ and $Q(t)$ from Equation (2.48) yields Equation (2.55) in the theorem statement. Equation (2.58) is derived from the following inequalities

$$\frac{P(t)^{1/2k}}{\varepsilon^{1/2k}} \le \frac{C_{\text{Trot}}^{(2k)}(H,t,\varepsilon)}{L} \frac{(1-q_B)^{1/2k}}{\Upsilon^{1-1/2k}}$$

$$\frac{Q(t)}{\varepsilon} = \Upsilon \frac{C_{\text{QD}}(H,t,\varepsilon)}{N_B} \left(\frac{\|h_B\|}{\|h\|}\right)^2. \tag{2.67}$$

These two inequalities are straightforward substitutions by plugging in results from the product formula costs in Theorem 2.3.1.1 and Theorem 2.3.2.1 into Equation (2.48), along with the definition of q_B. □

2.5.2 Conditions for Improvement

Now that we have bounded the Composite channel error and computed the query cost we ask the natural question: "When is a Composite channel better than just using Trotter?" Our first answer to this question will be analytic and can be found in Theorem 2.5.2.1 and are summarized in Table 1. We will be able to derive asymptotic conditions when a fixed partitioning can outperform either Trotter or QDrift. One issue that arises when making these comparisons is that we are comparing a Composite channel to *two* different simulation methods, each with their own query cost. The relative performance of Trotter to QDrift is dependent on the simulation time t and the error ε. It turns out that the ratio $\frac{C_{\text{QD}}}{C_{\text{Trot}}}$ depends on a power of the ratio $\frac{t}{\varepsilon}$. For very accurate and long simulations we observe that Trotter has superior cost, but whenever error requirements are not high or the simulation time is relatively short QDrift is the more efficient simulation method. One thought experiment to illuminate this is the limit as $t \ll 1$, in which case QDrift can replicate the exact time evolution statistics with a single sample whereas Trotter-Suzuki methods need to implement one operator exponential per term of the Hamiltonian.

Theorem 2.5.2.1 (Conditions for Composite Channel Improvements): *Let H be a family of Hamiltonians along with a partitioning scheme to generate a partition $H = A + B$ that varies with L. For a simulation time t and diamond distance error bound of ε, let ξ be the number such that $C_{\text{QD}}^{\xi} = C_{\text{Trott}}^{(2k)}$. There exists asymptotic regimes for the parameters L_A, $\|h_B\|$, and N_B such that*

$$C_{\text{Comp}}^{(2k)} \in o\Big(\min\Big\{C_{\text{Trot}}^{(2k)}, C_{\text{QD}}\Big\}\Big), \tag{2.68}$$

outlined below.

For the case when $C_{\text{Trott}}^{(2k)} < C_{\text{QD}}$, corresponding to $0 < \xi < 1$, if the following are satisfied

1. *$L_A(1-q_B)^{1/2k} \in o(L)$,*
2. *$\|h_B\| \in o\left(\|h\|^{\xi}\left(\frac{\sqrt{\varepsilon}}{t}\right)^{1-\xi}\right)$,*
3. *$N_B \in \Omega(L_A)$ and $N_B \in o\left(\frac{L}{(1-q_B)^{1/2k}}\right)$,*

then we have that $C_{\text{Comp}}^{(2k)} \in o\Big(C_{\text{Trot}}^{(2k)}\Big) = o\Big(\min\Big\{C_{\text{Trot}}^{(2k)}, C_{\text{QD}}\Big\}\Big)$.

For the case when $C_{\text{QD}} \le C_{\text{Trot}}^{(2k)}$, corresponding to $\xi \ge 1$, if the following are satisfied

1.
$$L_A \in o\left(L^{\xi}\left(\frac{t^{(2k+1)(\xi-1)}}{\varepsilon^{\xi-1}}\frac{\alpha_C(H)^{\xi}}{\alpha_C(A)+\alpha_C(\{A,B\})}\right)^{1/2k}\right), \tag{2.69}$$

2. *$\|h_B\| \in o(\|h\|)$,*
3. *and $N_B \in \Theta(L_A)$,*

then we have $C_{\text{Comp}}^{(2k)} \in o\big(C_{\text{QD}}\big) = o\Big(\min\Big\{C_{\text{Trot}}^{(2k)}, C_{\text{QD}}\Big\}\Big)$.

Note that for $\xi = 1$ the conditions on L_A and $\|h_B\|$ are the same in both cases: $L_A \in o(L)$ and $\|h_B\| \in o(\|h\|)$. The conditions on N_B are satisfied by $N_B \in \Theta(L_A)$ as the condition $N_B \in o\left(\frac{L}{(1-q_B)^{1/2k}}\right)$ is not valid when $\xi = 1$.

Proof: We start with the expression for the Composite channel cost from Theorem 2.5.1.1 repeated here for clarity

$$C_{\text{Comp}}^{(2k)} \le \Upsilon(\Upsilon L_A + N_B)\left[\frac{C_{\text{Trot}}^{(2k)}(H,t,\varepsilon)}{\Upsilon^{1-1/2k}}\frac{(1-q_B)^{1/2k}}{L} + C_{\text{QD}}(H,t,\varepsilon)\frac{\Upsilon}{N_B}\left(\frac{\|h_B\|}{\|h\|}\right)^2\right]. \tag{2.70}$$

We will split our analysis up into the two cases outlined in the theorem statement.

- $0 < \xi < 1$

In this scenario we have that $C_{\text{QD}} > C_{\text{Trot}}^{(2k)}$, so we can pull out the $C_{\text{Trot}}^{(2k)}$ cost and parametrize the ratio $\frac{C_{\text{QD}}}{C_{\text{Trot}}^{(2k)}} = C_{\text{QD}}^{1-\xi}$.

$$C_{\text{Comp}}^{(2k)} \leq C_{\text{Trot}}^{(2k)} \Upsilon(\Upsilon L_A + N_B) \left(\frac{(1-q_B)^{1/2k}}{\Upsilon^{1-1/2k} L} + C_{\text{QD}}(H,t,\varepsilon)^{1-\xi} \frac{\Upsilon}{N_B} \left(\frac{\|h_B\|}{\|h\|} \right)^2 \right). \tag{2.71}$$

We can then show that $C_{\text{Comp}}^{(2k)} \in o\left(C_{\text{Trot}}^{(2k)}\right)$ if we are able to show that every term in the expansion of the above two factors are $o(1)$. We do so term by term.

$$L_A \in o\left(\frac{L}{(1-q_B)^{1/2k}} \right) \implies \Upsilon^{1+1/2k} \frac{(1-q_B)^{1/2k} L_A}{L} \in o(1),$$

$$\|h_B\| \in o\left(\|h\|^{\frac{\xi}{2}} \left(\frac{\sqrt{\varepsilon}}{t} \right)^{1-\xi} \right) \implies \Upsilon^2 \left(\frac{\|h_B\|}{\|h\|} \right)^2 C_{\text{QD}}^{1-\xi} = \left(\frac{t^2}{\varepsilon} \right)^{1-\xi} \frac{\|h_B\|^2}{\|h\|^{\xi}} \in o(1),$$

$$N_B \in o\left(\frac{L}{(1-q_B)^{1/2k}} \right) \implies \Upsilon^{1/2k} (1-q_B)^{1/2k} \frac{N_B}{L} \in o(1). \tag{2.72}$$

The last term we have is

$$\Upsilon^3 C_{\text{QD}}^{1-\xi} \frac{\|h_B\|^2}{\|h\|^2} \frac{L_A}{N_B}. \tag{2.73}$$

Using the QDrift cost expression we have that $C_{\text{QD}}^{1-\xi} \leq 4^{1-\xi} \left(\frac{t^2}{\varepsilon} \right)^{1-\xi} \|h\|^{2(1-\xi)}$. This tells us that $C_{\text{QD}}^{1-\xi} \frac{\|h_B\|^2}{\|h\|^2} \in o(1)$ given the assumption $\|h_B\| \in o\left(\|h\|^{\frac{\xi}{2}} \left(\frac{\sqrt{\varepsilon}}{t} \right)^{1-\xi} \right)$. The total term above is then in $o(1)$ given the assumption that $N_B \in \Omega(L_A)$. As this is the last term in the expansion we have completed the $0 < \xi < 1$ case.

- $\xi \geq 1$

In this scenario we have $\frac{C_{\text{Trot}}^{(2k)}}{C_{\text{QD}}} = \left(C_{\text{Trot}}^{(2k)} \right)^{1-\xi}$ which allows us to write

$$C_{\text{Comp}}^{(2k)} \leq C_{\text{QD}} \Upsilon(\Upsilon L_A + N_B) \left(\frac{\left(C_{\text{Trot}}^{(2k)} \right)^{1-\xi}}{\Upsilon^{1-1/2k}} \frac{(1-q_B)^{1/2k}}{L} + \frac{\Upsilon}{N_B} \left(\frac{\|h_B\|}{\|h\|} \right)^2 \right). \tag{2.74}$$

We will tackle the hardest term in this expansion first, which is the one involving $\left(C_{\text{Trot}}^{(2k)} \right)^{1-\xi}$. Using the cost expression for Trotter given in Theorem 2.3.1.1 we have

$$\begin{aligned} &\Upsilon^{1+1/2k} (1-q_B)^{1/2k} \frac{L_A}{L} \left(C_{\text{Trot}}^{(2k)} \right)^{1-\xi} \\ &= \Upsilon^{2+1/2k} \left(\frac{t^{1+1/2k}}{\varepsilon^{1/2k}} \right)^{1-\xi} \frac{L_A}{L^{\xi}} \left(\frac{\alpha_C(A) + \alpha_C(\{A,B\})}{\alpha_C(H)^{\xi}} \right)^{1/2k}. \end{aligned} \tag{2.75}$$

This expression is in $o(1)$ given Assumption 1 from the theorem statement. We can then reduce the term involving N_B and $C_{\text{Trot}}^{(2k)}$ to the previous term as $N_B \in \Theta(L_A)$

$$\Upsilon N_B \frac{\left(C_{\text{Trot}}^{(2k)}\right)^{1-\xi}}{\Upsilon^{1-1/2k}} \frac{(1-q_B)^{1/2k}}{L} \in \Theta\left((1-q_B)^{1/2k} \frac{L_A}{L} \left(C_{\text{Trot}}^{(2k)}\right)^{1-\xi}\right) \in o(1), \tag{2.76}$$

where the last inclusion was shown for the previous term. The last two terms in the expansion involving the spectral norms are as follows

$$\|h_B\| \in o(\|h\|) \implies \Upsilon^2 \left(\frac{\|h_B\|}{\|h\|}\right)^2 \in o(1)$$

$$\|h_B\| \in o(\|h\|) \text{ and } N_B \in \Theta(L_A) \implies \Upsilon \frac{L_A}{N_B} \left(\frac{\|h_B\|}{\|h\|}\right)^2 \in o(1). \tag{2.77}$$

As we have shown all four terms in the expansion are $o(1)$ we have that $C_{\text{Comp}}^{(2k)} \in o\left(C_{\text{QD}}\right) = o\left(\min\left\{C_{\text{QD}}, C_{\text{Trot}}^{(2k)}\right\}\right)$ for $0 < \xi < 1$ which completes the proof. □

Now that we have concrete bounds we would like to build some intuition for the assumptions that go into the theorem. As the expressions become fairly unwieldy in the generic setting we can isolate the scenario where we expect the most benefit from using a Composite framework, which is when $C_{\text{QD}} = C_{\text{Trot}}^{(2k)}$ or $\xi = 1$. The rationale behind this intuition is that if $C_{\text{QD}} \ll C_{\text{Trot}}^{(2k)}$, we can imagine building a composite channel by starting with a solely QDrift partitioning scheme and then moving over the most advantageous terms to the Trotter partition. We have a lot less room until the costs of Trotter begin to add up. Similar logic holds for the $C_{\text{QD}} \gg C_{\text{Trot}}^{(2k)}$ regime. In the intermediate regime $C_{\text{QD}} \approx C_{\text{Trot}}^{(2k)}$ we have a bit more flexibility to move terms around without bumping in to these costly partitions.

Another benefit to analyzing the $\xi = 1$ scenario is that the resulting assumptions needed for cost improvements simplify significantly. The three requirements reduce to the following:

1. $L_A \in o(L)$, which we use the simplification that $\alpha_{\text{comm}}(H) \geq \alpha_{\text{comm}}(A) + \alpha_{\text{comm}}(\{A, B\})$ implies that $L_A \in o(L)$ is sufficient to meet the exact requirement in Theorem 2.5.1.1,
2. $\|h_B\| \in o(\|h\|)$,
3. and $N_B \in \Theta(L_A)$.

These convey much more intuition than the generic conditions we proved. Simply put these conditions say that if you can find a partitioning that contains most of the spectral weight of the Hamiltonian in a small number of terms then then the resulting Composite channel will be asymptotically cheaper than using a single simulation method. This of course only holds rigorously at the ratio of $\frac{t}{\varepsilon}$ such that Trotter and QDrift costs are equal, but we will demonstrate numerically in Section 2.6 that these advantages hold in nearby values of t and ε. To summarize this section we provide the following table that contains the requirements in Theorem 2.5.2.1 but in a easier to read format.

	$C_{\text{QD}} \leq C_{\text{Trot}}^{(2k)} \iff \xi \geq 1$	$C_{\text{QD}} > C_{\text{Trot}}^{(2k)} \iff 0 < \xi < 1$
$L_A \in$	$o\left(\frac{L}{(1-q_B)^{1/2k}}\right)$	$o\left(L^{\xi}\left(\frac{t^{1+1/2k}}{\varepsilon^{1/2k}}\right)^{\xi-1}\frac{\alpha_C(H)^{\xi/2k}}{(\alpha_C(A)+\alpha_C(\{A,B\}))^{1/2k}}\right)$
$\lVert h_B \rVert \in$	$o\left(\lVert h \rVert^{\xi}\left(\frac{t}{\sqrt{\varepsilon}}\right)^{\xi-1}\right)$	$o(\lVert h \rVert)$
(Lower Bound) $N_B \in$	$\Omega(L_A)$	$\Omega(L_A)$
(Upper Bound) $N_B \in$	$o\left(\frac{L}{(1-q_B)^{1/2k}}\right)$	$O(L_A)$

Table 1: Summary of asymptotic requirements for parameters of interest when $C_{\text{QD}}^{\xi} = C_{\text{Trot}}^{(2k)}$ to yield $C_{\text{Comp}}^{(2k)} \in o\left(\min\left\{C_{\text{QD}}, C_{\text{Trot}}^{(2k)}\right\}\right)$.

2.5.3 Partitioning Scheme

As we have seen throughout, the partition used to create a Composite channel has a significant impact on the resulting number of operator exponentials needed. This makes partitioning an important problem, but one that is also fairly challenging as the solution space is 2^L. In this section we show how a simple partitioning scheme called `chop` can create partitions that work exceptionally well for systems with large separations between the largest spectral norm terms and the smallest spectral norms. `chop` creates a partition $A + B$ of a Hamiltonian given a norm cutoff h_{chop}, all terms with spectral norm above h_{chop} are placed into A and all those below are placed into QDrift:

$$A_{\texttt{chop}} := \sum_{i=1}^{L} \mathbb{I}\left[h_i \geq h_{\text{chop}}\right] h_i H_i, \qquad B_{\texttt{chop}} := \sum_{i=1}^{L} \mathbb{I}\left[h_i < h_{\text{chop}}\right] h_i H_i, \tag{2.78}$$

where we use $\mathbb{I}[\text{Proposition}]$ to denote the standard indicator function where $\mathbb{I}[\text{True}] = 1, \mathbb{I}[\text{False}] = 0$.

`chop` will prove to be a very useful partitioning scheme both analytically and numerically. Analytically we will be able to show that it satisfies the conditions outlined in Theorem 2.5.2.1 for specific Hamiltonians. Numerically, it is very simple to create a specified partition from a Hamiltonian and further it is straightforward to optimize as the partition can be adjusted with a a single parameter h_{chop}. This still leaves open the problem of choosing the right number of QDrift samples N_B, but we did not find this parameter an issue to optimize analytically or numerically.

Theorem 2.5.3.1 (Simulation Improvements for Exponentially Decaying Hamiltonians): *Let H be a Hamiltonian $H = \sum_i h_i H_i$ such that the spectral norms decay exponentially $h_i = 2^{-i}$ and the commutator structure is small, i.e. $\alpha_{\text{comm}}(H, 2k) \in O\left(\frac{\log L}{L}\right)$. Then the* `chop` *partitioning scheme that places the largest $\log L$ terms into Trotter and the remaining terms into QDrift satisfy the conditions for asymptotic improvement outlined in Theorem 2.5.2.1 whenever $\frac{t}{\varepsilon}$ is at the crossover point where $C_{\text{QD}} = C_{\text{Trot}}^{(2k)}$. This requires setting the number of QDrift samples N_B to $\log L$ as well.*

Proof Sketch: As the conditions for improvement depend on L_A, $\|h_B\|$, and N_B, but we know that $L_A = N_B = \log(L)$, all we need to compute is $\|h_B\|$. This is done using straightforward sums:

$$\|h_B\| = \sum_{i=\log(L)+1}^{L} 2^{-i} = 2^{1-(\log(L)+1)} - 2^{-L} = \frac{1}{L} - 2^{-L} \in \Theta(L^{-1}). \tag{2.79}$$

The total norm can be computed similarly

$$\|h\| = \sum_{i=0}^{L-1} 2^{-i} = 1 - 2^{-L} \in \Theta(1). \tag{2.80}$$

Now we just need to check the conditions on each parameter. We will analyze the requirements for ξ for each parameter instead of doing a case by case analysis for the two regimes of ξ.

Starting with N_B, we find that $N_B = L_A \in \Omega(L_A)$ trivially and that $N_B = \log(L) \in o\left(\frac{L}{(1-q_B)^{1/2k}}\right)$ along with $N_B = L_A \in O(L_A)$ guarantee that N_B meets the conditions in Theorem 2.5.2.1 straightforwardly.

We then turn to the next simplest parameter $\|h_B\|$. For $0 < \xi < 1$ we require $\|h_B\| \in \|h\|$, and since we computed that $\|h_B\| \in \Theta(L^{-1})$ and $\|h\| \in \Theta(1)$ this condition holds. For $\xi \geq 1$ we require $\|h_B\| \in o\left(\|h\|^{\xi}\left(\frac{t}{\sqrt{\varepsilon}}\right)^{\xi-1}\right)$. If we require that $t > \sqrt{\varepsilon}$ then we can guarantee this holds as

$$\|h_B\| \in \Theta(L^{-1}) \in o(1) = o\left(\|h\|^{\xi}\left(\frac{t}{\sqrt{\varepsilon}}\right)^{\xi-1}\right). \tag{2.81}$$

The last term we will need to address is L_A. For $\xi \geq 1$ we require $L_A \in o(L)$, which is trivially satisfied. The second condition is that $L_A \in$

We will summarize the conditions needed whenever $\xi = 1$, which was discussed informally after the proof of Theorem 2.5.2.1. We need that $L_A \in o(L)$, $\|h_B\| \in o(\|h\|)$, and $N_B \in \Theta(L_A)$. First we note that we can satisfy two of these conditions from the start by setting $L_A = \log L = N_B$. The only remaining condition is that $\|h_B\| \in o(\|h\|)$. This can be computed via straightforward summations

$$\|h\| = \sum_{i=0}^{L-1} 2^{-i} = 1 - 2^{-L}$$

$$\|h_B\| = \sum_{i=\log(L)+1}^{L} 2^{-i} = 2^{1-(\log(L)+1)} - 2^{-L} = L^{-1} - 2^{-L}. \tag{2.82}$$

As $\|h\| \in \Theta(1)$ and $\|h_B\| \in \Theta(L^{-1})$ we have that $\|h_B\| \in \|h\|$. Therefore the conditions for asymptotic improvement at $\xi = 1$ are met fairly easily. □

2.6 Numerics

This section is primarily based on joint work conducted in [10]. In that paper we provided analytic extensions of the real time composite channels developed in [35] in addition to conducting a numeric investigation to the performance of Composite channels, however we will only discuss the real time composite simulation results. We study various chemistry and condensed matter Hamiltonians used to benchmark simulation methods, namely Hydrogen chain systems, Uniform Electron Gas (Jellium), and nearest neighbor spin models on graphs. We find that both heuristic partitioning schemes developed by hand and partitionings based on machine learning optimizers perform comparably. Both partitioning schemes are capable of finding partitions that have anywhere from 1/2 to 1/19 lower query costs at the crossover times, depending on the Hamiltonian being studied. A summary of the numeric results are contained below in Table 2.

Hamiltonian	ξ	**Num. Terms**	*** - Time**
Hydrogen-3	2.3	62	Real -
5 Site Jellium	9.2	56	Real -
6 Site Jellium	18.8	94	Real -
7 Site Jellium	10.4	197	Real -
7 Spin Graph	4.1	49	Real -
8 Spin Graph	3.9	64	Real -
8 Spin Heisenberg	3.1	29	Imag. -
Hydrogen-3	2.3	62	Imag. -
6 Site Jellium	18.8	94	Imag. -

Table 2: Summary of gate cost improvements observed via the *crossover advantage* t_ξ defined in Equation (contingent on optimization convergence). We observe that savings tend to somewhat improve as the number of terms increases (within the same model), with the exception of Jellium 7 where optimizer struggles with partitioning due to the number of terms. This is evident in the lack of monotonicity of $C\left(\widetilde{\mathcal{C}}^1\right)$ in Figure ref{fig:Jellium56}. The most significant savings are seen for the Jellium models. Even in cases where the number of terms are comparable to other models, larger advantages are persistent in Jellium. This further establishes the spectral norm distribution as one of the most important indicators of performance in the composite framework.

2.7 Discussion

Chapter 3

Preparing Thermal Quantum States

Now that we have techniques for simulating time-independent Hamiltonian evolution with randomized terms we turn to utilizing these methods for preparing thermal states.

Chapter 4

Appendix

Chapter

Bibliography

[1] R. M. Neal, Probabilistic inference using Markov chain Monte Carlo methods, (1993)

[2] J. M. Deutsch, Eigenstate Thermalization Hypothesis, Reports on Progress in Physics **81**, 82001 (2018)

[3] E. B. Davies, Markovian master equations, Communications in Mathematical Physics **39**, 91 (1974)

[4] C.-F. Chen, M. J. Kastoryano, F. G. S. L. Brandão, and A. Gilyén, Quantum Thermal State Preparation, (2023)

[5] A. Gilyén, C.-F. Chen, J. F. Doriguello, and M. J. Kastoryano, Quantum generalizations of Glauber and Metropolis dynamics, (2024)

[6] Z. Ding, B. Li, and L. Lin, Efficient quantum Gibbs samplers with Kubo–Martin–Schwinger detailed balance condition, (2024)

[7] T. S. Cubitt, Dissipative ground state preparation and the Dissipative Quantum Eigensolver, (2023)

[8] M. Hagan and N. Wiebe, Composite Quantum Simulations, Arxiv Preprint Arxiv:2206.06409 (2022)

[9] M. Hagan and N. Wiebe, The Thermodynamic Cost of Ignorance: Thermal State Preparation with One Ancilla Qubit, Arxiv Preprint Arxiv:2502.03410 (2025)

[10] M. Pocrnic, M. Hagan, J. Carrasquilla, D. Segal, and N. Wiebe, Composite Qdrift-product formulas for quantum and classical simulations in real and imaginary time, Physical Review Research **6**, 13224 (2024)

[11] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, Simulation of electronic structure Hamiltonians using quantum computers, Molecular Physics **109**, 735 (2011)

[12] S. P. Jordan, K. S. Lee, and J. Preskill, Quantum algorithms for quantum field theories, Science **336**, 1130 (2012)

[13] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, Proceedings of the National Academy of Sciences **114**, 7555 (2017)

[14] R. Babbush, D. W. Berry, and H. Neven, Quantum simulation of the Sachdev-Ye-Kitaev model by asymmetric qubitization, Phys. Rev. A **99**, 40301 (2019)

[15] Y. Su, D. W. Berry, N. Wiebe, N. Rubin, and R. Babbush, Fault-Tolerant Quantum Simulations of Chemistry in First Quantization, PRX Quantum **2**, 40332 (2021)

[16] T. E. O'Brien et al., Efficient quantum computation of molecular forces and other energy gradients, Phys. Rev. Res. **4**, 43210 (2022)

[17] D. Aharonov and A. Ta-Shma, *Adiabatic Quantum State Generation and Statistical Zero Knowledge*, in *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing* (2003), pp. 20–29

[18] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, Communications in Mathematical Physics **270**, 359 (2007)

[19] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian Dynamics with a Truncated Taylor Series, Phys. Rev. Lett. **114**, 90502 (2015)

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

[21] G. H. Low and I. L. Chuang, Hamiltonian Simulation by Qubitization, Quantum **3**, 163 (2019)

[22] G. H. Low, V. Kliuchnikov, and N. Wiebe, Well-conditioned multiproduct Hamiltonian simulation, Arxiv Preprint Arxiv:1907.11679 (2019)

[23] G. H. Low and N. Wiebe, Hamiltonian Simulation in the Interaction Picture, Arxiv Preprint Arxiv:1897.10070 (2019)

[24] E. Campbell, Random Compiler for Fast Hamiltonian Simulation, Phys. Rev. Lett. **123**, 70503 (2019)

[25] N. Wiebe, D. Berry, P. Høyer, and B. C. Sanders, Higher order decompositions of ordered operator exponentials, Journal of Physics A: Mathematical and Theoretical **43**, 65203 (2010)

[26] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, Theory of Trotter Error with Commutator Scaling, Phys. Rev. X **11**, 11020 (2021)

[27] D. W. Berry, A. M. Childs, Y. Su, X. Wang, and N. Wiebe, Time-dependent Hamiltonian simulation with L^1-norm scaling, Quantum **4**, 254 (2020)

[28] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, Gate-count estimates for performing quantum chemistry on small quantum computers, Physical Review a **90**, (2014)

[29] D. Poulin, M. B. Hastings, D. Wecker, N. Wiebe, A. C. Doherty, and M. Troyer, The Trotter step size required for accurate quantum simulation of quantum chemistry, Arxiv Preprint Arxiv:1406.4920 (2014)

[30] I. D. Kivlichan, C. E. Granade, and N. Wiebe, Phase estimation with randomized Hamiltonians, Arxiv Preprint Arxiv:1907.10070 (2019)

[31] A. Rajput, A. Roggero, and N. Wiebe, Hybridized Methods for Quantum Simulation in the Interaction Picture, Quantum **6**, 780 (2022)

[32] Y. Ouyang, D. R. White, and E. T. Campbell, Compilation by stochastic Hamiltonian sparsification, Quantum **4**, 235 (2020)

[33] S. Jin and X. Li, A Partially Random Trotter Algorithm for Quantum Hamiltonian Simulations, Arxiv Preprint Arxiv:2109.07987 (2021)

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

[35] M. Hagan and N. Wiebe, Composite quantum simulations, Quantum **7**, 1181 (2023)