Randomized Hamiltonian Compilation

Christopher Kang

June 2020

Abstract

Hamiltonian simulation has enormous utility for modeling complex chemical and physical systems. The major technique is Trotterization, which leverages the Trotter-Suzuki formulas to accurately approximate the Hamiltonian with performance asymptotically better than classical algorithms. However, Trotterization vastly exceeds the available resources of near-term quantum chips. We derive bounds for randomized Hamiltonian compilation [1] and compare them to Trotterization of different orders.

1 Introduction to Trotterization

1.1 Problem: Hamiltonian Simulation

The aim of Hamiltonian simulation is modeling a system's evolution over time.

Problem 1. *We are given the following parameters.*

1. $|\psi_0\rangle$*, the initial state*
2. H*, the Hamiltonian*
3. t*, the number of time evolutions for simulation*

How can we efficiently simulate:

$$e^{-iHt} |\psi_0\rangle = |\psi(t)\rangle \tag{1}$$

With minimal error?

For $t << 1$, there are some immediate steps that can be performed. Assume that we can decompose H:

$$H = \sum_{j}^{L} h_j H_l \tag{2}$$

Where H_l Hermitian with a maximum singular value of 1 and $h_j \in \mathbb{C}$. Thus, we may separate e^{iHt}:

$$e^{iHt} = \prod_{j=1}^{L} e^{ih_j H_l t} + O(L^2 t^2) \tag{3}$$

This follows because H_i, H_j typically do not anticommute, incurring an error that grows quickly as t grows. as t gets large, this error quickly becomes unacceptable.

1.2 Trotterization

For $t << 1$, the error term of the prior formula is negligble. However, we are often interested in simulating larger t. The Trotter-Suzuki formulas aim to allow for simulation when t large by segmenting the time steps into smaller pieces.

We 'slice' the exponential into N terms:

$$e^{iHt} = e^{iHtN/N} = \Big(\prod_{j=1}^{L} e^{ih_j H_j t/N}\Big)^N + O\Big(\frac{(t\Lambda L)^2}{N}\Big) \tag{4}$$

Where $\Lambda = \max_j h_j$. So, say we desire an error ϵ. Then, we'd like:

$$\begin{aligned} \frac{(t\Lambda L)^2}{N} &\leq \epsilon \\ N &\geq \frac{(t\Lambda L)^2}{\epsilon} \end{aligned} \tag{5}$$

We can then refine this restriction of N into an estimate for the gate count [3]:

Fact 1 (Gate counts)**.** *1st order deterministic Trotter scales in*

$$O\Big(\frac{L^3(\Lambda t)^2}{\epsilon}\Big) \tag{6}$$

2nd order deterministic Trotter scales in:

$$O\Big(\frac{L^{5/2}(\Lambda t)^{3/2}}{e^{1/2}}\Big) \tag{7}$$

(2k)th order probabilistic Trotter scales in:

$$O\Big(\frac{L^2(\Lambda t)^{1+1/2k}}{\epsilon^{1/2k}}\Big) \tag{8}$$

Notice that even with these advanced techniques [2], the scaling is still quadratic in L. This motivates the core question: are there ways to reduce the scaling in L, thereby allowing for reduced cost in simulating complex Hamiltonians?

2 Randomized Compilers for Hamiltonians

2.1 Algorithm: QDRIFT

Random compilation is a pre-processing technique used to optimize Trotterization. Instead of directly applying the $h_j H_j$ sequentially, we randomly select and apply some number of Trotterized terms. We then discover that the gate count has better scaling with relation to L than typical Trotterization, at the cost of scaling with respect to t.

Algorithm 1: QDRIFT

```
**Result:** Evolution $e^{iHt}$ applied to $\rho$ with some error $\epsilon$
Decompose Hamiltonian into $\sum_{j=1}^{L} h_j H_j$ where $h_j \in \mathbb{R}$;
Create $\lambda = \sum h_j$;
Randomly sample $N$ values from $[1, L]$, each with probability $\frac{h_j}{\lambda}$;
Apply the $e^{i\lambda H_j/N}$ exponential for every sample value above;
```

We begin by assuming that our Hamiltonian has been decomposed into a sum of normalized Hermitian operators H_j, similar to in Equation (2). However, specifically note that we constrain our coefficients $h_j \in \mathbb{R}$. From these coefficients, we create an auxiliary variable λ:

$$\lambda = \sum h_j \tag{9}$$

Now, to generate the final list of operators, create a distribution where the probability of selecting H_j is weighted by its leading coefficient:

$$\mathbb{P}(U = H_j) = p_j = \frac{h_j}{\lambda} \tag{10}$$

Sample from this distribution N times, producing a list of values $\mathbf{k} = \{k_1, k_2..\}$. Thus, the final Hamiltonian components to simulate $V_{\mathbf{k}}$ are:

$$V_{\mathbf{k}} = \prod_{j=1}^{N} e^{i\tau_j H_{k_j}} \tag{11}$$

Where τ_j is a special weighting.

2.2 Identifying τ

Before we continue further analysis, we must identify suitable values for τ. We must compare the evolution of QDRIFT ($\mathcal{E}(\rho)$) to the original Trotter formula ($\mathcal{U}(\rho)$). So, let us apply one term in V_k and the $e^{iHt/N}$ term, and evaluate the effects:

Theorem 2.1 (Optimal τ). *When we select τ as follows:*

$$\tau = \frac{t\lambda}{N}$$

The error between the QDRIFT approximation $\mathcal{E}(\rho)$ and the original Trotterized formula $\mathcal{U}(\rho)$ is in the second order with respect to t.

Proof. Let's model the effect of this Hamiltonian on the quantum state. Because this instance is probabilistic, we choose to use the density operator. Consider our starting state to be $|\psi_0\rangle$, so $\rho = |\psi_0\rangle \langle\psi_0|$. We can model the evolution of the quantum state by probabilistically applying one term of $V_{\mathbf{k}}$:

$$\mathcal{E}(\rho) = \sum_j p_j e^{i\tau H_j} \rho e^{-i\tau H_j} \tag{12}$$

In comparison, we were interested in applying $e^{iHt/N}$, which would have yielded the following density operator:

$$\mathcal{U}_N(\rho) = e^{iHt/N}\rho e^{-iHt/N} \tag{13}$$

We'd like to compare $\mathcal{E}(\rho)$ to $\mathcal{U}_N$ to identify how close our approximation is. Recall:

Fact 2 (Matrix Exponentials)**.** *For any square matrix X, we define its exponential e^X as:*

$$e^X = \sum_{k=0}^{\infty} \frac{1}{k!} X^k$$

This exponential is well-defined for all matrices X.

Thus, expanding $\mathcal{E}(\rho)$:

$$\begin{aligned}
\sum_j p_j e^{i\tau H_j}\rho e^{-i\tau H_j} &= \sum_j \frac{h_j}{\lambda} e^{i\tau H_j}\rho e^{-\tau H_j} \\
&= \sum_j \frac{h_j}{\lambda}\Big(\sum_{k=0}^{\infty} \frac{(i\tau H_j)^k}{k!}\Big)\rho\Big(\sum_{k=0}^{\infty} \frac{(-i\tau H_j)^k}{k!}\Big) \\
&= \sum_j \frac{h_j}{\lambda}(1 + i\tau H_j + O(\tau^2))\rho(1 - i\tau H_j + O(\tau^2)) \\
&= \sum_j \frac{h_j}{\lambda}(\rho + i\tau H_j\rho + O(\tau^2)(1 - i\tau H_j + O(\tau^2)) \\
&= \sum_j \frac{h_j}{\lambda}(\rho + i\tau H_j\rho - i\tau\rho H_j + O(\tau^2)) \\
&= \rho + i\sum_j \frac{\tau h_j}{\lambda}(H_j\rho - \rho H_j) + O(\tau^2)
\end{aligned} \tag{14}$$

The channel we wanted to simulate $\mathcal{U}_N(\rho) = e^{iHt/N}\rho e^{-iHt/N}$. Again, using the expansion:

$$\begin{aligned}
\mathcal{U}_N(\rho) &= e^{iHt/N}\rho e^{-iHt/N} \\
&= \Big(\sum_{k=0}^{\infty} \frac{(iHt)^k}{N^k k!}\Big)\rho\Big(\sum_{k=0}^{\infty} \frac{(-iHt)^k}{N^k k!}\Big) \\
&= \Big(1 + \frac{iHt}{N} + O\Big(\frac{t^2}{N^2}\Big)\Big)\rho\Big(1 - \frac{iHt}{N} + O\Big(\frac{t^2}{N^2}\Big)\Big) \\
&= \Big(\rho + \frac{it}{N}H\rho + O\Big(\frac{t^2}{N^2}\Big)\Big)\Big(1 - \frac{iHt}{N} + O\Big(\frac{t^2}{N^2}\Big)\Big) \\
&= \rho - \frac{it}{N}\rho H + \frac{it}{N}H\rho + O\Big(\frac{t^2}{N^2}\Big) \\
&= \rho + \frac{it}{N}(H\rho - \rho H) + O\Big(\frac{t^2}{N^2}\Big)
\end{aligned} \tag{15}$$

Because $H = \sum_j h_j H_j$:

$$\begin{aligned}
\mathcal{U}_N(\rho) &= \rho + \frac{it}{N}(\sum_j h_j H_j\rho - \rho\sum_j h_j H_r) + O\Big(\frac{t^2}{N^2}\Big) \\
&= \rho + i\sum_j \frac{th_j}{N}(H_j\rho - \rho H_j) + O\Big(\frac{t^2}{N^2}\Big)
\end{aligned} \tag{16}$$

As noted in the paper, we'd like to identify where the difference in the density matrices $\mathcal{E}(\rho) - \mathcal{U}(\rho)$ has zeroth/first order terms which vanish. Therefore, we'd like:

$$i \sum_j \frac{\tau h_j}{\lambda}(H_j\rho - \rho H_j) = i \sum_j \frac{t h_j}{N}(H_j\rho - \rho H_j) \tag{17}$$

Thus, we'd need $\frac{\tau}{\lambda} = \frac{t}{N}$, or:

$$\tau = \frac{t\lambda}{N} \tag{18}$$

As desired. □

2.3 Bounding Higher Order Terms

Now that we have a value for τ, we'd like to identify an upper bound for $\|\mathcal{E}(\rho) - \mathcal{U}_N(\rho)\|$. We can then use this bound to model the differences of the entire simulation, or $\left\|\mathcal{E}^N(\rho) - \mathcal{U}(\rho)\right\|$.

First, let's manipulate our model of the Hamiltonian on the density operator. We introduce the Liouvillian superoperator:

Definition 1 (Liouvillian Superoperator)**.** *A* ***Liouvillian superoperator*** *takes an operator and returns another [4]. For example,*

$$\mathcal{L}(\rho) = i(H\rho - \rho H)$$

So that,

$$\mathcal{L}^2(\rho) = i(Hi(H\rho - \rho H) - i(H\rho - \rho H)H) = -HH\rho + H\rho H + H\rho H - \rho HH = -H^2\rho + 2H\rho H - \rho H^2$$

So, we can actually leverage it to demonstrate a rearrangement of our Hamiltonian simulation upon our density matrix:

Theorem 2.2 (Liouvillian Representation)**.** *Given the following Liouvillian superoperator:*

$$\mathcal{L}(n) = i(H\rho - \rho H)$$

We can represent the unitary channel $\mathcal{E}$ as:

$$\mathcal{E} = e^{iHt}\rho e^{-iHt} = e^{t\mathcal{L}}(\rho) = \sum_{k=0}^{\infty} \frac{t^k \mathcal{L}^k(\rho)}{k!}$$

Proof. Recall the matrix exponential forms of e^{iHt}, e^{-iHt}:

$$e^{iHt} = \sum_{k=0}^{\infty} \frac{i^k t^k H^k}{k!} \tag{19}$$

$$e^{-iHt} = \sum_{k=0}^{\infty} \frac{(-1)^k i^k t^k H^k}{k!} \tag{20}$$

So, let's consider the form this takes. Instead, take each term of e^{iHt} as a_k and each term of e^{-iHt} as b_k. If we consider $e^{iHt}\rho e^{-iHt}$ to be:

$$(a_0 + a_1 + ... + a_n)\rho(b_0 + b_1 + ... + b_n)$$

Where $n \to \infty$, then this is really an n by n matrix P whose values $P_{jk} = a_j \rho b_k$. Furthermore, we can actually express b_k in terms of a_k:

$$b_k = \begin{cases} a_k & k \text{ even} \\ -a_k & k \text{ odd} \end{cases}$$

By the definition of e^{-iHt}. Thus, we define the general term of P_{jk} as:

$$P_{jk} = \begin{cases} a_j \rho a_k & k \text{ even} \\ -a_j \rho a_k & k \text{ odd} \end{cases} = (-1)^k a_j \rho a_k$$

Now, let's substitute our value for $a_k = \frac{i^k t^k H^k}{k!}$. Thus:

$$\begin{aligned} P_{jk} &= (-1)^k \frac{i^j t^j H^j}{j!} \rho \frac{i^k t^k H^k}{k!} \\ &= (-1)^k \frac{i^{j+k} t^{j+k}}{j!k!} H^j \rho H^k \end{aligned} \tag{21}$$

Let's group these terms by $j+k$ constant. Visually, these would be the diagonals of the matrix. Let's designate C_l as the sum of the terms with t^l. Then,

$$\begin{aligned} C_l &= \sum_{k=0}^{l} (-1)^k t^l \frac{i^l}{k!(l-k)!} H^{l-k} \rho H^k \\ &= i^l t^l \sum_{k=0}^{l} (-1)^k \frac{l!}{k!(l-k)!} \frac{H^{l-k} \rho H^k}{l!} \\ &= i^l t^l \sum_{k=0}^{l} (-1)^k \binom{l}{k} \frac{H^{l-k} \rho H^k}{l!} \end{aligned} \tag{22}$$

Now, let's compare this to the form of the nth power Liouvillian. I claim:

Lemma 2.1. *The nth power Liouvillian takes the following form:*

$$\mathcal{L}^n(\rho) = i^n \sum_{k=0}^{n} (-1)^k \binom{n}{k} H^{n-k} \rho H^k$$

Proof. We'll show this holds by induction.

For the base case, recognize that we have demonstrated this for $n = 1$:

$$\mathcal{L}^1(\rho) = iH\rho - i\rho H$$

For the inductive hypothesis, assumes this holds for n. Then, we'd like to prove in the inductive step that this holds for $n+1$. So, manipulating $\mathcal{L}^{n+1}(\rho)$:

$$\begin{aligned} \mathcal{L}^{n+1}(\rho) &= i(H\mathcal{L}^n(\rho) - \mathcal{L}^n(\rho)H) \\ &= i(H\Big(i^n \sum_{k=0}^{n} (-1)^k \binom{n}{k} H^{n-k} \rho H^k\Big) - \Big(i^n \sum_{k=0}^{n} (-1)^k \binom{n}{k} H^{n-k} \rho H^k\Big)H) \\ &= i^{n+1} \Big[\sum_{k=0}^{n} (-1)^k \binom{n}{k} H^{n-k+1} \rho H^k - \sum_{k=0}^{n} (-1)^k \binom{n}{k} H^{n-k} \rho H^{k+1}\Big] \\ &= i^{n+1} \Big[\binom{n}{0} H^{n+1} \rho + \sum_{k=1}^{n} (-1)^k \binom{n}{k} H^{n-k+1} \rho H^k - \sum_{k=0}^{n-1} (-1)^k \binom{n}{k} H^{n-k} \rho H^{k+1} + (-1)^{n+1} \rho H^{n+1}\Big] \end{aligned}$$

Now, let's analyze the inner terms and see if they can be manipulated. Recognize, with a reindexing:

$$\begin{aligned}
&= \sum_{k=1}^{n}(-1)^k \binom{n}{k} H^{n-k+1}\rho H^k - \sum_{k=0}^{n-1}(-1)^k \binom{n}{k} H^{n-k}\rho H^{k+1} \\
&= \sum_{k=1}^{n}(-1)^k \binom{n}{k} H^{n-k+1}\rho H^k + \sum_{k=1}^{n}(-1)^k \binom{n}{k-1} H^{n-k+1}\rho H^k \qquad (23)
\end{aligned}$$

Furthermore, recall that:

$$\binom{a}{b} + \binom{a}{b-1} = \binom{a+1}{b}$$

So,

$$\sum_{k=1}^{n}(-1)^k \binom{n}{k} H^{n-k+1}\rho H^k + \sum_{k=1}^{n}(-1)^k \binom{n}{k-1} H^{n-k+1}\rho H^k = \sum_{k=1}^{n}(-1)^k \binom{n+1}{k} H^{n-k+1}\rho H^k$$

Thus, substituting this into our original equation, we see:

$$\begin{aligned}
&= i^{n+1}\left[H^{n+1}\rho + \sum_{k=1}^{n}(-1)^k \binom{n+1}{k} H^{n-k+1}\rho H^k + (-1)^{n+1}\rho H^{n+1}\right] \\
&= i^{n+1}\sum_{k=0}^{n+1}(-1)^k \binom{n+1}{k} H^{n-k+1}\rho H^k
\end{aligned}$$

Thus,

$$\mathcal{L}^{n+1}(\rho) = i^{n+1}\sum_{k=0}^{n+1}(-1)^k \binom{n}{k} H^{n-k+1}\rho H^k \qquad (24)$$

So, by induction, we've demonstrated the formula for the nth Liouvillian, as desired. □

Now, by Lemma 2.1, analyze the lth term of the sequence of $e^{i\mathcal{L}(\rho)}$:

$$\frac{t^l\mathcal{L}^l}{l!} = \frac{i^l t^l}{l!}\sum_{k=0}^{l}(-1)^k \binom{l}{k} H^{l-k}\rho H^k \qquad (25)$$

By Equation (22), C_l is:

$$C_l = \frac{i^l t^l}{l!}\sum_{k=0}^{l}(-1)^k \binom{l}{k} H^{l-k}\rho H^k$$

Thus, $C_l = \frac{t^l\mathcal{L}^l}{l!}$. So:

$$e^{iHt}\rho e^{-iHt} = \sum_{l=0}^{\infty} C_l = \sum_{k=0}^{\infty}\frac{t^k\mathcal{L}^k(\rho)}{k!} \qquad (26)$$

As desired. □

Now that we may represent both $\mathcal{E}, \mathcal{U}_N$ in convenient forms, we need effective measure of their distance. We introduce some common norms:

Definition 2 (1-Norm)**.** *Given some matrix A, it has a 1-norm defined by:*

$$\|A\|_1 = \mathrm{Tr}[\sqrt{\mathrm{A}^\dagger \mathrm{A}}]$$

Alternatively, the 1-Norm is the sum of the singular values of A.

Definition 3 (Infty-Norm)**.** *Given some matrix A, it has an ∞-norm defined by:*

$$\|A\| = \inf\{c \geq 0 : \|Av\| \leq c\,\|v\|\,, v \in W\}$$

Where W is the range vector space. Alternatively, the norm is the largest singular value of A.

Definition 4 (Diamond Norm)**.** *Given a superoperator $\mathcal{P}$, the diamond norm of $\mathcal{P}$ is defined as:*

$$\|\mathcal{P}\|_\diamond := \sup_{\rho;\|\rho\|_1=1} \|(\mathcal{P} \otimes \mathbb{1})(\rho)\|_1$$

Finally, we define a metric for the distance of the channels:

Definition 5 (Diamond distance)**.** *The diamond distance of two channels $\mathcal{A}, \mathcal{B}$ is defined as:*

$$d_\diamond(\mathcal{A}, \mathcal{B}) = \frac{1}{2}\|\mathcal{A} - \mathcal{B}\|_\diamond$$

Now that we've established some definitions, we can also begin to bound elements of our representation. Recognize that the diamond norm has some simple bounds when considering the Liouvillian:

Lemma 2.2.

$$\|\mathcal{L}\|_\diamond \leq 2\|H\| \leq 2\lambda$$

Proof. Recognize by definition that:

$$\|\mathcal{L}\|_\diamond = \sup_{\rho;\|\rho\|_1=1} \|(\mathcal{L} \otimes \mathbb{1})(\rho)\|_1 \tag{27}$$

Equivalently, recalling that H, ρ Hermitian and $\|A\|_1 = \sqrt{A^\dagger A}$:

$$\sup_{\rho;\|\rho\|_1=1} \|H\rho - \rho H\|_1 = \sup_{\rho;\|\rho\|_1=1} \mathrm{Tr}[H\rho - \rho H] \tag{28}$$

Then, by the Von-Neumann inequality, we know:

$$\mathrm{Tr}[H\rho] \leq \|H\| \,\mathrm{Tr}\,\rho = \|H\| \tag{29}$$

So:

$$\|\mathcal{L}\|_\diamond \leq 2\|H\| \tag{30}$$

Finally, we observe:

$$2\|H\| \leq 2\lambda \tag{31}$$

Because we are operating with L2 norms, the Schatten-∞ norm actually seeks the largest singular value, which we've set to λ by Equation (2). Thus,

$$\|\mathcal{L}\|_\diamond \leq 2\|H\| \leq 2\lambda \tag{32}$$

As desired. □

Recognize that we can separate the Liouvillian so that:

$$\mathcal{L} = \sum_j h_j \mathcal{L}_j$$

By the decomposition of the Hamiltonian in Equation (2). Thus, we can also identify bounds on these Liouvillian components:

Lemma 2.3.

$$\left\|\mathcal{L}_j\right\|_\diamond \leq 2\left\|H_j\right\| \leq 2$$

Proof. We can apply a similar technique as Lemma 2.2; simply note that because we chose H_j Hermitian and h_j so that the maximum singular value was 1, operator norm is at most 1, as desired. □

Now that we have some fundamental definitions, we aim to demonstrate:

Theorem 2.3. *Suppose $\tau = \frac{t\lambda}{N}$. Given $\mathcal{E} = \sum_j p_j e^{i\tau H_j t} \rho e^{-i\tau H t}$ and $\mathcal{U}_N = e^{iHt/N} \rho e^{-iHt/N}$, the difference in the two channels $\delta = d_\diamond(\mathcal{E}, \mathcal{U}_N)$ will be bounded by:*

$$\delta \leq \frac{2\lambda^2 t^2}{N^2} e^{2\lambda t/N} \approx \frac{2\lambda^2 t^2}{N^2}$$

For modest N.

Proof. By Theorem 2.2, we can represent $\mathcal{E}$ in terms of its Liouvillian:

$$\mathcal{E} = \sum_j p_j e^{i\tau H_j t} \rho e^{-i\tau H_j t} = \sum_j p_j e^{\tau \mathcal{L}_j} = \sum_j \frac{h_j}{\lambda} e^{\tau \mathcal{L}_j} \tag{33}$$

In contrast, our default channel is:

$$\mathcal{U}_N = e^{iHt/N} \rho e^{-iHt/N} = e^{t\mathcal{L}/N} \tag{34}$$

Now that we have Liouvillian representations for both of these channels, let's express the first few terms of each.

$$\begin{aligned}
\mathcal{E} &= \sum_j \frac{h_j}{\lambda} \Big(\sum_{k=0}^{\infty} \frac{\tau^k \mathcal{L}_j^k}{k!} \Big) \\
&= \sum_j \frac{h_j}{\lambda} \Big(\mathbb{1} + \tau \mathcal{L}_j + \sum_{k=2}^{\infty} \frac{\tau^k \mathcal{L}_j^k}{k!} \Big) \\
&= \mathbb{1} + \sum_j \frac{h_j}{\lambda} \tau \mathcal{L}_j + \sum_j \frac{h_j}{\lambda} \sum_{k=2}^{\infty} \frac{\tau^k \mathcal{L}_j^k}{k!} \\
&= \mathbb{1} + \frac{\tau}{\lambda} \mathcal{L} + \sum_j \frac{h_j}{\lambda} \sum_{k=2}^{\infty} \frac{\tau^k \mathcal{L}_j^k}{k!}
\end{aligned} \tag{35}$$

Also, recall by hypothesis that $\tau = \frac{t\lambda}{N}$. Thus,

$$\mathcal{E} = \mathbb{1} + \frac{t}{N} \mathcal{L} + \sum_j \frac{h_j}{\lambda} \sum_{k=2}^{\infty} \frac{t^k \lambda^k \mathcal{L}_j^k}{k! N^k} \tag{36}$$

In contrast, the channel of the desired channel is:

$$\begin{aligned}\mathcal{U}_N &= e^{t\mathcal{L}/N}\\ &= \sum_{k=0}^{\infty}\frac{t^k\mathcal{L}^k}{k!N^k}\\ &= \mathbb{1} + \frac{t}{N}\mathcal{L} + \sum_{k=2}^{\infty}\frac{t^k\mathcal{L}^k}{k!N^k}\end{aligned} \tag{37}$$

Now, with these representations, we can create a bound on their diamond-norm difference. Consider:

$$\begin{aligned}\|\mathcal{U}_N - \mathcal{E}\|_\diamond &= \left\|\sum_{k=2}^{\infty}\frac{t^k\mathcal{L}^k}{k!N^k} - \sum_j \frac{h_j}{\lambda}\sum_{k=2}^{\infty}\frac{t^k\lambda^k\mathcal{L}_j^k}{k!N^k}\right\|_\diamond\\ &\leq \left\|\sum_{k=2}^{\infty}\frac{t^k\mathcal{L}^k}{k!N^k}\right\|_\diamond + \left\|\sum_j \frac{h_j}{\lambda}\sum_{k=2}^{\infty}\frac{t^k\lambda^k\mathcal{L}_j^k}{k!N^k}\right\|_\diamond\end{aligned} \tag{38}$$

By the Triangle Inequality. Now, by homogeneity:

$$= \sum_{k=2}^{\infty}\frac{t^k\left\|\mathcal{L}^k\right\|_\diamond}{k!N^k} + \sum_j \frac{h_j}{\lambda}\sum_{k=2}^{\infty}\frac{t^k\lambda^k\left\|\mathcal{L}_j^k\right\|_\diamond}{k!N^k} \tag{39}$$

Let's bound the Liouvillian terms. Recognize that we can find upper bounds on both $\mathcal{L}, \mathcal{L}_j$:

Lemma 2.4 (Upper Bound on Liouvillian Representation)**.**

$$\|\mathcal{L}^n\|_\diamond \leq (2\lambda)^n$$

Proof. By sub-multiplicativity of the diamond norm, we know that:

$$\|\mathcal{L}^n\|_\diamond \leq \|\mathcal{L}\|_\diamond^n$$

Furthermore, by Lemma 2.2, we know that:

$$\|\mathcal{L}\|_\diamond \leq 2\lambda$$

Thus,

$$\|\mathcal{L}^n\|_\diamond \leq (2\lambda)^n$$

As desired. □

Similarly, we can bound the Liouvillian components:

Lemma 2.5 (Upper Bound on Liouvillian Components)**.**

$$\left\|\mathcal{L}_j^n\right\|_\diamond \leq 2^n$$

Proof. Again, by submultiplicativity, we have:

$$\left\|\mathcal{L}_j^n\right\|_\diamond \leq \|\mathcal{L}_j\|_\diamond^n$$

By Lemma 2.3, we have that:

$$\|\mathcal{L}_j\|_\diamond \leq 2$$

Thus,

$$\left\|\mathcal{L}_j^n\right\|_\diamond \leq 2^n$$

As desired. □

Now, by Lemma 2.4 and Lemma 2.5, we know that:

$$\begin{aligned}\|\mathcal{U}_N - \mathcal{E}\| &\leq \sum_{k=2}^{\infty} \frac{t^k(2\lambda)^k}{k!N^k} + \sum_j \frac{h_j}{\lambda} \sum_{k=2}^{\infty} \frac{t^k \lambda^k 2^k}{k!N^k} \\ &= \left(1 + \sum_j \frac{h_j}{\lambda}\right) \sum_{k=2}^{\infty} \frac{t^k(2\lambda)^k}{k!N^k} \\ &= 2\sum_{k=2}^{\infty} \frac{1}{k!}\left(\frac{2\lambda t}{N}\right)^k \qquad (40)\end{aligned}$$

Recall that we define the diamond norm to be double the distance, thus:

$$d(\mathcal{U}_N, \mathcal{E}) \leq \sum_{k=2}^{\infty} \frac{1}{k!}\left(\frac{2\lambda t}{N}\right)^k \qquad (41)$$

Now, leveraging this tail bound:

Fact 3.

$$\sum_{k=2}^{\infty} \frac{x^n}{n!} \leq \frac{x^2}{2} e^x$$

We yield a bound of:

$$d(\mathcal{U}_N, \mathcal{E}) \leq \frac{1}{2}\left(\frac{2\lambda t}{N}\right)^2 e^{2\lambda t/N} = \frac{2\lambda^2 t^2}{N^2} e^{2\lambda t/N} \approx \frac{2\lambda^2 t^2}{N^2} \qquad (42)$$

As desired. □

We then extend Theorem 2.3 to model the behavior of QDRIFT on the full Hamiltonian:

Theorem 2.4.

$$d_\diamond(\mathcal{E}^N, \mathcal{U}) \leq \frac{2\lambda^2 t^2}{N} e^{2\lambda t/N} \approx \frac{2\lambda^2 t^2}{N}$$

Again given modest N.

Proof. Because the diamond norm is subadditive under composition, we yield:

$$\begin{aligned}d(\mathcal{U}, \mathcal{E}^N) &\leq N d(\mathcal{U}_N, \mathcal{E}) \\ &= \frac{2\lambda^2 t^2}{N} e^{2\lambda t/N} \qquad (43) \\ &\approx \frac{2\lambda^2 t^2}{N} \qquad (44)\end{aligned}$$

Because $2\lambda t/N \approx 0$ for large N. Thus, the error bound holds, as desired. □

This error analysis culminates in Theorem 2.4, a significant result that dictates the overall error incurred by using QDRIFT over typical Hamiltonian simulation methods. By representing our Hamiltonian with Liouvillians and creating upper bounds for the superoperators, we compactly identified upper bounds for the distance between the channels.

2.4 Evaluating the Impact of QDRIFT

Now that we have error bounds that scale in the size of N, we are interested in the gate counts required to obtain adequate performance. We demonstrate simple gate bounds:

Theorem 2.5. *QDRIFT performs in* $O(\frac{\lambda^2 t^2}{\epsilon})$ *time.*

Proof. Suppose we are given some $\epsilon > 0$ maximum error. Then, the N required is:

$$\begin{aligned} \epsilon &\approx \frac{2\lambda^2 t^2}{N} \\ N &= \frac{2\lambda^2 t^2}{\epsilon} \in O\Big(\frac{\lambda^2 t^2}{\epsilon}\Big) \end{aligned} \tag{45}$$

As desired. □

Now that we have an asymptotic bound, let's compare it to Trotterization. Recall that $\Lambda = \max_j h_j$ and $\lambda = \sum h_j$. If many h_j large, then $\lambda \approx \Lambda L$, so by Theorem 2.5:

Corollary 2.1. *For* $\lambda \approx \Lambda L$, *QDRIFT scales in:*

$$O\Big(\frac{L^2(\Lambda t)^2}{\epsilon}\Big) \tag{46}$$

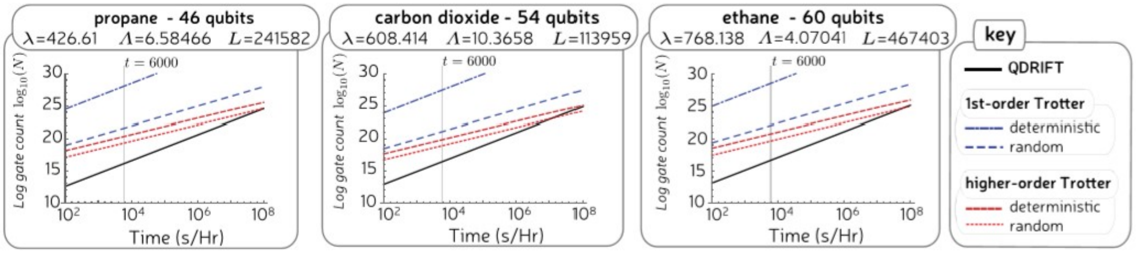
If most $h_j \ll \Lambda$, then we obtain a different bound:

Corollary 2.2. *For* $\lambda \approx \Lambda\sqrt{L}$, *QDRIFT scales in:*

$$O\Big(\frac{L(\Lambda t)^2}{\epsilon}\Big) \tag{47}$$

Referencing Corollary 2.1, Corollary 2.2, it's apparent that QDRIFT performs well when Λ relatively small. Recalling Trotterization's scaling from Fact 1, QDRIFT scales better than both 1st/2nd order Trotter in terms of L, meaning that more complex Hamiltonians may perform better with randomized compilation.

However, the tradeoff is a quadratic dependence upon t, whereas the best Trotter algorithms require nearly linear complexity. So, when comparing QDRIFT and Trotter on the same Hamiltonian, there will be some t where Trotter has less gate complexity. Experimental data indicates that this t exceeds typical values. Consider the following graphs on propane, carbon dioxide, and ethane [1]:



As predicted, the slope of QDRIFT's line is steeper than the best Trotter algorithms, representing asymptotically worse behavior in t. However, real-world applications typically choose $t = 6000$ (the vertical line). At this t, each of the molecules has better performance with QDRIFT than with any of the Trotter techniques. Thus, in regimes with moderate to small t, QDRIFT could reduce the cost of Hamiltonian simulation problems.

3 Early Experimental Results

3.1 Data

As an extension of the material, I also briefly implemented the randomized protocol in Q# (link here). To test the implementation, every bond length from 0.2 to 2.85 (step size 0.05) was fed to the state prep algorithm. The size of the new compiled Hamiltonian was set to 5; the original term has a size of 5 ($L = 5, N = 5$). The Trotterized or compiled Hamiltonian was then applied, and phase estimation used to identify the energy estimate. Each bond length had 10 trials to account for the probabilistic nature of both phase estimation and randomized compilation. Some early experimental results are posted below:

Figure 1 compares the average energy estimates versus bond length. This result demonstrates the increased variability that emerges from the randomization - the compiled algorithm (blue) is much more volatile than the Trotterized algorithm (orange). However, it's important to note that the randomized compilation still enables the identification of the ground bond length (0.7).

Figure 2 compares the median energy estimates versus bond length. Again, the compiled algorithm (blue) exhibits greater variability than the Trotterized algorithm (orange), though this is also influenced by the variability of individual trials.

Figure 3 compares the min/max/median energy estimates for both the compiled and original algorithms vs bond length. The variability of the compiled algorithm is pronounced in the graph, though it's noticeable that even the original algorithm has significant variability that is incurred by failures in the phase estimation.

3.2 Discussion and Applicability

As implied earlier, the strength of compilation exists when L large, as gate count has better scaling. In the case of H2, $L = 5$ is hardly large. So, we expect this error bound to narrow as L grows; similarly, we expect gate count to fall relative to the uncompiled algorithm.

We must also consider potential errors. One source is precision - Robust Phase Estimation, the technique used for phase estimation, has a qubit/gate requirement that scales with phase precision. Furthermore, the technique is inherently probabilistic. These are only two potential sources of error that accentuate the variability of the compiled Hamiltonian, but a physical chip could be subject to a multitude of other errors.

Another factor to consider is gate counts. Due to both technical and time limitations, this was **not implemented concurrently to the accuracy analysis**. Again, **this analysis is preliminary and should be considered as such**. To compare gate costs, we compare the differences in the number of rotation gates necessary (ignoring CNOTs, Paulis, and other gates). Q#'s estimates indicate that the number of rotation gate counts are equivalent for $N = L$; thus, as L increases, we suspect that the scaling will continue to improve for the compiled Hamiltonian, as the necessary N will be less compared to the uncompiled algorithm. Further analysis on Hamiltonians with other term types, like PQRS and PQQR terms, could also yield interesting results with respect to gate costs [5].

4 Conclusion

This paper explored Earl Campbell's Randomized Hamiltonian Compilation method ('QDRIFT'). Randomized compilation operates by probablistically selecting the component H_j to simulate, instead of simulating all of H. By finding appropriate τ for tuning the $e^{i\tau H_j}$ gates, we were able to compare the quantum channels of the randomly compiled circuit versus the fully Trotterized circuit. Then, we demonstrated asymptotic bounds that show better scaling with respect to a Hamiltonian's complexity, with the tradeoff of a greater dependence on simulation time.

After demonstrating theoretical tradeoffs, we performed a brief implementation of QDRIFT in Q#. While the resultants were quite variable, they show that QDRIFT is within the general neighborhood of the uncompiled algorithm. Further research is necessary to identify the performance of QDRIFT on physical devices and/or with larger Hamiltonians.

While QDRIFT does not address the underlying complexity of Hamiltonian simulations and challenges for near-term devices, this technique reduces gate counts for complex simulations. Furthermore, it suggests the potential for additional pre-processing to improve performance, even in the presence of asymptotic tradeoffs.

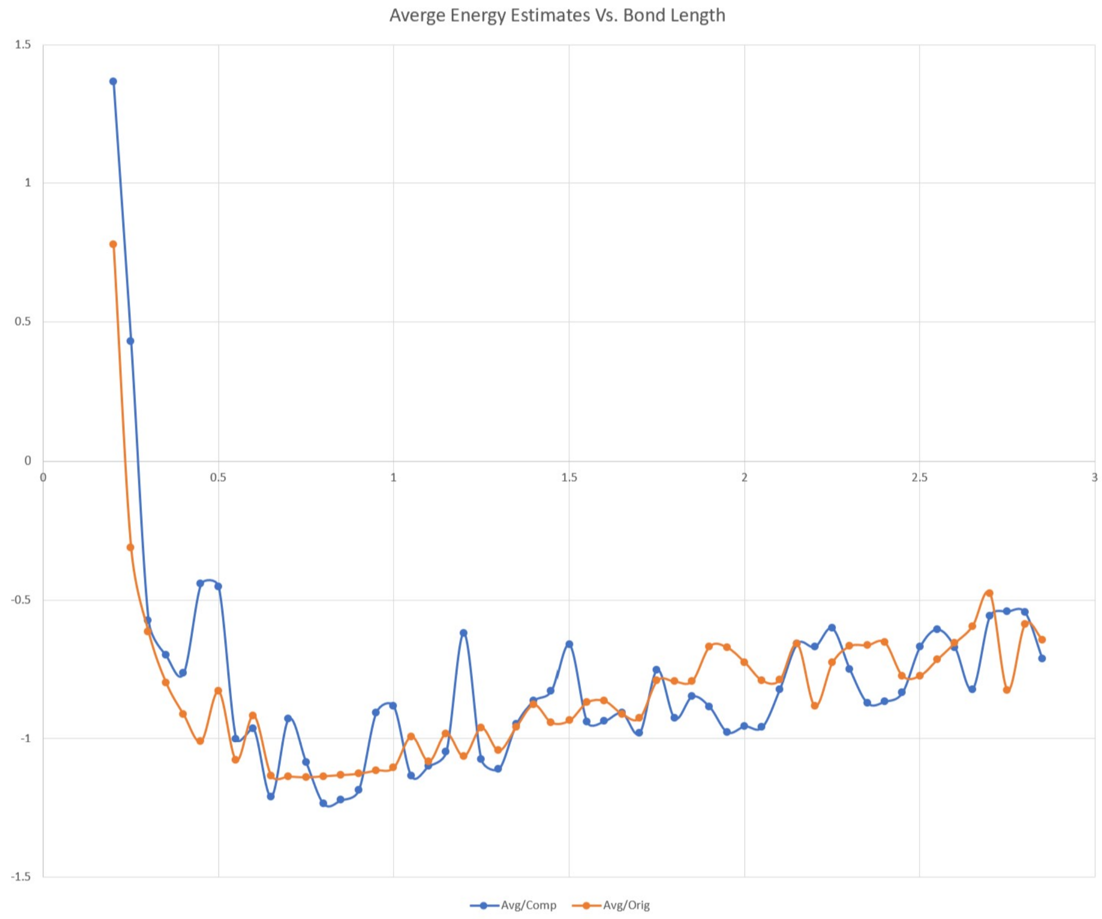


Figure 1: Average Energy Estimates vs Bond Length over Compiled and Uncompiled Hamiltonians

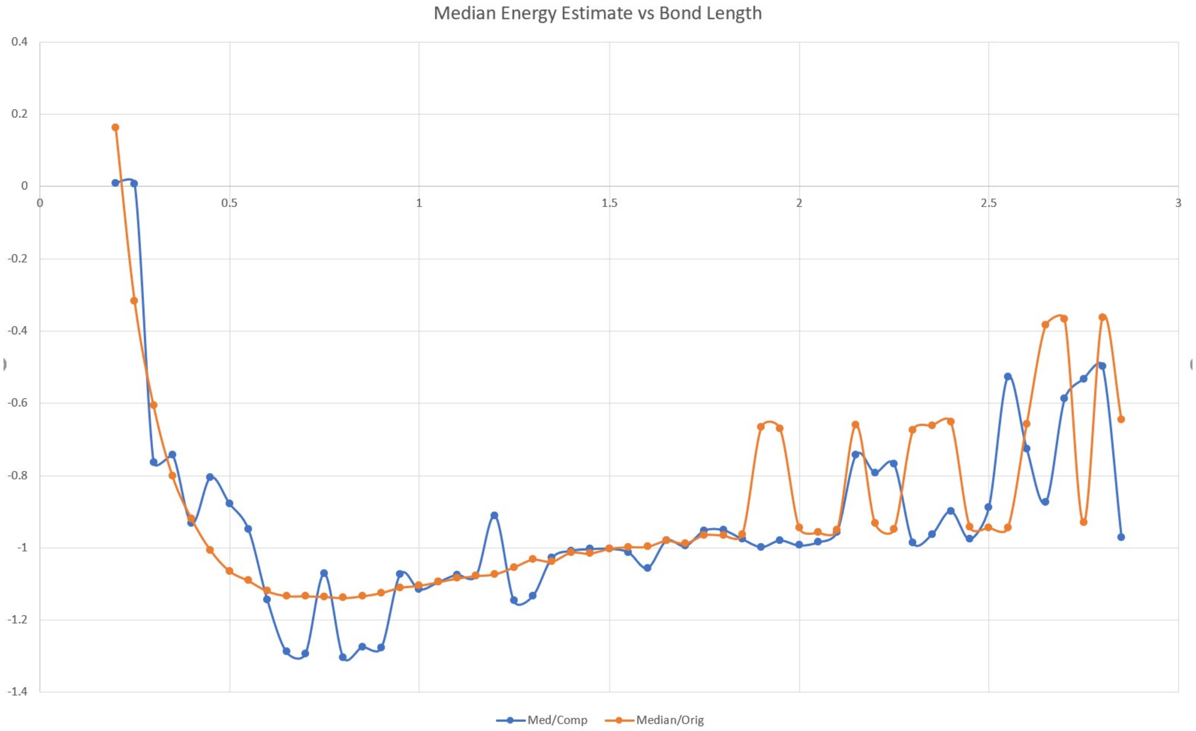


Figure 2: Median Energy Estimates vs Bond Length over Compiled and Uncompiled Hamiltonians

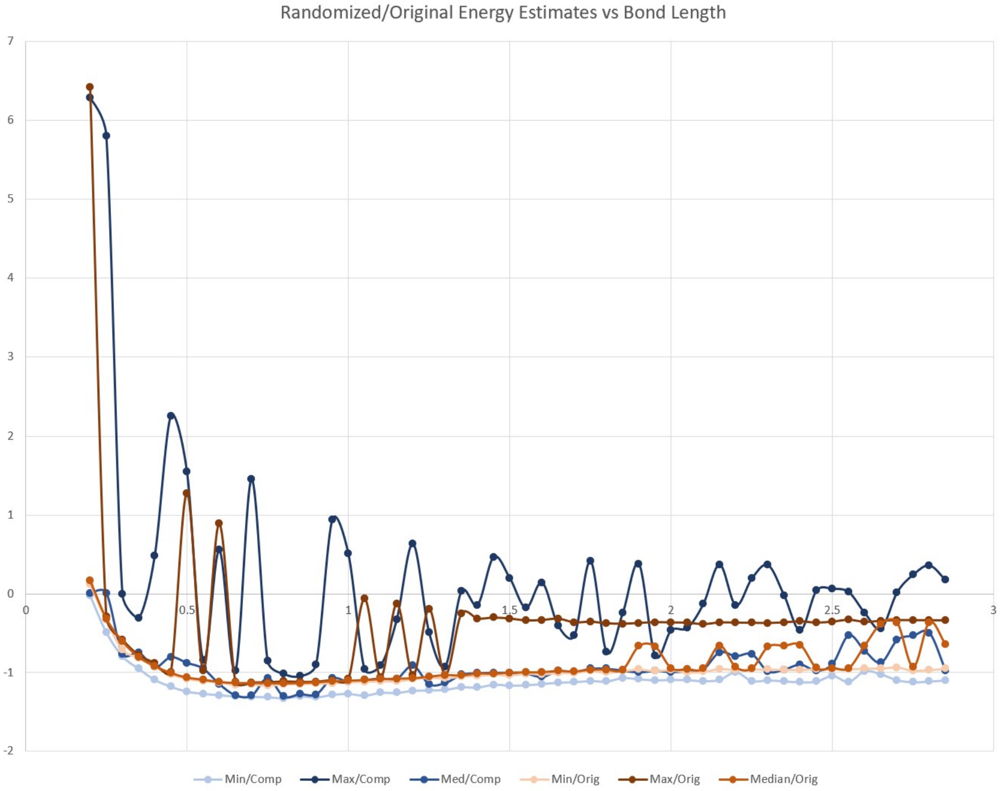


Figure 3: Min/Med/Max Energy Estimates vs Bond Length over Compiled and Uncompiled Hamiltonians

References

[1] Earl Campbell. “Random Compiler for Fast Hamiltonian Simulation”. In: *Phys. Rev. Lett.* 123 (7 Aug. 2019), p. 070503. DOI: 10.1103/PhysRevLett.123.070503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.070503.

[2] Andrew M. Childs, Aaron Ostrander, and Yuan Su. “Faster quantum simulation by randomization”. In: *Quantum* 3 (Sept. 2019), p. 182. ISSN: 2521-327X. DOI: 10.22331/q-2019-09-02-182. URL: http://dx.doi.org/10.22331/q-2019-09-02-182.

[3] Microsoft. *Trotter-Suzuki Formulas*. URL: https://docs.microsoft.com/en-us/quantum/libraries/chemistry/concepts/algorithms.

[4] Mark E. Tuckerman. *Time evolution of the density operator*. URL: http://www.nyu.edu/classes/tuckerman/stat.mechII/lectures/lecture_11/node3.html.

[5] James D. Whitfield, Jacob Biamonte, and Alán Aspuru-Guzik. “Simulation of electronic structure Hamiltonians using quantum computers”. In: *Molecular Physics* 109.5 (Mar. 2011), pp. 735–750. ISSN: 1362-3028. DOI: 10.1080/00268976.2011.552441. URL: http://dx.doi.org/10.1080/00268976.2011.552441.