Exploiting anticommutation in Hamiltonian simulation

Qi Zhao[1] and Xiao Yuan[2,3]

[1] Joint Center for Quantum Information and Computer Science, University of Maryland, College Park, Maryland 20742, USA

[2] Center on Frontiers of Computing Studies, Department of Computer Science, Peking University, Beijing 100871, China

[3] Stanford Institute for Theoretical Physics, Stanford University, Stanford California 94305, USA

Quantum computing can efficiently simulate Hamiltonian dynamics of many-body quantum physics, a task that is generally intractable with classical computers. The hardness lies at the ubiquitous anticommutative relations of quantum operators, in corresponding with the notorious negative sign problem in classical simulation. Intuitively, Hamiltonians with more commutative terms are also easier to simulate on a quantum computer, and anti-commutative relations generally cause more errors, such as in the product formula method. Here, we theoretically explore the role of anti-commutative relation in Hamiltonian simulation. We find that, contrary to our intuition, anticommutative relations could also reduce the hardness of Hamiltonian simulation. Specifically, Hamiltonians with mutually anti-commutative terms are easy to simulate, as what happens with ones consisting of mutually commutative terms. Such a property is further utilized to reduce the algorithmic error or the gate complexity in the truncated Taylor series quantum algorithm for general problems. Moreover, we propose two modified linear combinations of unitaries methods tailored for Hamiltonians with different degrees of anti-commutation. We numerically verify that the proposed methods exploiting anti-commutative relations could significantly improve the simulation accuracy of electronic Hamiltonians. Our work sheds light on the roles of commutative and anti-commutative relations in simulating quantum systems.

Qi Zhao: zhaoq@umd.edu
Xiao Yuan: xiaoyuan@pku.edu.cn

1 Introduction

It is a notoriously hard problem to classically simulate an arbitrary large many-body physics systems. As proposed by Feynman in 1982 [20], quantum computers can directly encode a quantum system and hence efficiently simulate any quantum Hamiltonian dynamics. With the help of quantum computers, various quantum algorithms have been to proposed [5, 6, 8, 25, 26] and applied for studying different problems such as spin models [30], fermionic lattice models [36], quantum chemistry [2, 3, 35], and quantum field theories [23]. For a given Hamiltonian $H = \sum_{l=1}^{L} \alpha_l H_l$ with coefficients α_l and Pauli operators H_l, we focus on the simulation of the time evolution operator $U_0(t) = e^{-iHt}$, which is crucial for studying both the dynamic and static properties [1, 24, 27]. We consider the product formula methods [32] and ones based on linear-combination-of-unitaries (LCU) [16]. They have shown different asymptotical gate complexities with respect to the simulation time t, the accuracy ε, and the property of H.

For the product formula method [13–15, 17, 18, 33], also known as the Trotter-Suzuki algorithm, it divides the evolution time t into small time steps δt and approximates each $e^{-iH\delta t}$ via a product of $e^{-i\alpha_l H_l \delta t}$, for example as

$$e^{-iH\delta t} \approx \prod_l e^{-i\alpha_l H_l \delta t} + \mathcal{O}(\delta t^2).$$

The error term $\mathcal{O}(\delta t^2)$ is a pessimistic worst-case estimation, and it could be dramatically improved by considering commutative relations between different H_l. For example, when all H_l are mutually commutative, the approximation becomes accurate and the error vanishes. We refer to Refs. [14, 17, 19, 21, 31, 34] for error analyses of product formula methods for more general problems. We note that due to the simplicity of the product formula method, it is widely

used in experimental implementations of Hamiltonian simulation with near-term quantum devices [4, 11, 12].

It is intuitively straightforward to see that the commutative relation could simplify or ease the complexity of Hamiltonian simulation whereas the anti-commutative relation could make the problem hard. While this seems to be true for the product formula method where Hamiltonians consisting of mutually (anti-)commutative terms have zero (maximal) approximation errors, the intuition breaks down in the LCU based methods. Specifically, we consider the truncated Taylor series algorithm [8], which, when expanding the operator $e^{-iH\delta t}$ to the Kth order, gives

$$e^{-iH\delta t} \approx \sum_{j=0}^{K} \frac{\left(\sum_l -i\alpha_l H_l \delta t\right)^j}{j!} + \mathcal{O}(\delta t^{K+1}).$$

The first summation term is realized via LCU and the approximation error comes from the truncated Taylor expansion error. It is not hard to see that the commutative relation could not universally reduce the approximation error as it does in the product formula method. Instead, the anti-commutative relation may cause cancellation of the higher order approximation error and hence be exploited to increase the simulation accuracy.

In this work, we investigate the role of anti-commutative relation in Hamiltonian simulation. We show that anti-commutative relation could also simplify the complexity of Hamiltonian simulation. We first use the anti-commutative relation to reduce the the truncated error in the Taylor series method, which lead to a tighter error analysis in contrast to the worse case one. Then we propose two modified LCU algorithms that exploit the anti-commutative relation and further reduce high-order truncation errors. Moreover, we show that Hamiltonians with mutually anti-commutative terms are perfectly simulable. We numerically test the performances of the proposed methods for electronic Hamiltonians and find significant improvements of the simulation accuracy with shallower quantum circuits. Our work thus broadens our understanding of commutative and anti-commutative relations in Hamiltonian simulation.

2 Preliminary

In this section, we review the product formula method and the Taylor series method. We show how the commutative relation plays an important role in the product formula method. In this work, we assume that the Hamiltonian H is expressed as

$$H = \sum_{l=1}^{L} \alpha_l H_l, \tag{1}$$

where H_l is a tensor product of Pauli operators and α_l are real coefficients. Each pair of elements H_i and H_j are either commutative or anti-commutative,

$$[H_i, H_j] = H_i H_j - H_j H_i = 0 \text{ (commutative)},$$
$$\{H_i, H_j\} = H_i H_j + H_j H_i = 0 \text{ (anti-commutative)}. \tag{2}$$

The target is to simulate the time evolution operator $U_0(t) = e^{-iHt}$ with quantum circuits.

2.1 Product formula method

The product formula (PF) approach approximates $U_0(t)$ via the product of exponentials of the individual operators H_l. By splitting the evolution time t into r segments (r also known as the number of Trotter steps) and making r sufficiently large, we can ensure the simulation is arbitrarily precise. In the first order PF method, we approximate $U_0(t)$ via a product form,

$$\begin{aligned} U_0(t) &= e^{-i\sum_l \alpha_l H_l t} \\ &\approx U_1(t) \\ &= \left(\prod_{l=1}^{L} e^{-i\alpha_l H_l t/r}\right)^r. \end{aligned} \tag{3}$$

Recent error analyses [17, 19] showed that the approximation error is related to the commutative relation of the operators

$$\|U_1(t) - U_0(t)\| \leq \frac{t^2}{2r} \sum_{l_1=1}^{L} \left\| \sum_{l_2 \geq l_1}^{L} [\alpha_{l_2} H_{l_2}, \alpha_{l_1} H_{l_1}] \right\|, \tag{4}$$

where $\|\cdot\|$ corresponds to the spectral norm. The approximation error becomes smaller when more pairs of H_{l_1} and H_{l_2} commute, and the error vanishes when every term commutes to each other. Note that any two Pauli operators either commute or anti-commute, and thus the approximation error mainly originates from the anti-commutative relation.

2.2 Taylor series method

Now we review the truncated Taylor series (TS) method [8], which is a recently introduced more advanced Hamiltonian simulation algorithm that achieves a different asymptotic error scaling. The main idea is to apply the (truncated) Taylor series of the evolution operator and implement the procedure for LCU. We rewrite the evolution operation in the Taylor series,

$$U_0(t) = e^{-iHt} = \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!},$$

and the truncated Taylor series to the K-th order is

$$
\begin{aligned}
\widetilde{U}(t) &= \sum_{k=0}^{K} \frac{(-itH)^k}{k!}, \\
&= \sum_{k=0}^{K} \sum_{l_1,\cdots,l_k=1}^{L} \frac{t^k}{k!} \alpha_{l_1} \cdots \alpha_{l_k} (-i)^k H_{l_1} \cdots H_{l_k}, \\
&= \sum_{j=0}^{J} \beta_j V_j,
\end{aligned}
$$
(5)

which is a linear combination of unitaries $V_j = (-i)^k H_{l_1} \cdots H_{l_k}$ with the coefficients $\beta_j = \frac{t^k}{k!} \alpha_{l_1} \cdots \alpha_{l_k}$. Without loss of generality, here we set each coefficient $\alpha_l > 0$ and thus each β_j is positive.

We also need to divide evolution time t into r segments. In each segment, the algorithmic error of the approximation mainly comes from the finite truncated Taylor series

$$\|\widetilde{U}(t/r) - U_0(t/r)\| \le e^{\alpha t/r} \frac{(\alpha t/r)^{K+1}}{(K+1)!}. \quad (6)$$

where α denotes the summation of α_l, $\alpha = \sum_l \alpha_l$. Given a fixed total allowed error ε, the truncated order parameter K can be chosen as

$$K = O\left(\frac{\log \frac{\alpha t}{\varepsilon}}{\log \log \frac{\alpha t}{\varepsilon}} \right). \quad (7)$$

We refer to Appendix A and Ref.[8, 17] for details of the implementation and error analysis. From the above analysis, the commutative relation does not help much in reducing the truncation error. The truncation error exists even if the Hamiltonian consists of mutually commutative terms (as long as $H \ne 0$). In the next section, we show that it is actually the anti-commutative relation

that plays a more important role in determining the error of the Taylor series method. We further show that the anti-commutative relation can be applied to reduce the algorithmic error and design modified Taylor series methods.

3 Anti-commutative cancellation

We first show the general idea of anti-commutative cancellation in refining the error analysis. We expand the time evolution operator $U_0(t) = e^{-iHt}$ as

$$U_0(t) = \sum_{m=0}^{\infty} \frac{(-it)^m}{m!} H^m, \quad (8)$$

where

$$
\begin{aligned}
H^m &= \left(\sum_l \alpha_l H_l \right)^m \\
&= \sum_{l_1,\ldots,l_m=1}^{L} \alpha_{l_1} \alpha_{l_2} \ldots \alpha_{l_m} H_{l_1} H_{l_2} \ldots H_{l_m}.
\end{aligned}
$$
(9)

Each H^m has L^m terms and its spectral norm is bound by

$$\|H^m\| \le \left(\sum_{l=1}^{L} \alpha_l \right)^m = \alpha^m. \quad (10)$$

In the original error analysis, this upper bound α^m is used to estimate the truncation error, which is overestimated without considering the relations between different H_l. To refine the estimation of the norm $\|H^m\|$, we consider each sequence in Eq. (9)

$$H_{l_1} H_{l_2} \ldots H_{l_{m-1}} H_{l_m}, \quad (11)$$

where $l_1, \ldots l_m \in \{1, 2, \ldots, L\}$. If two terms, for instance H_{l_1} and H_{l_2} are anti-commutative with $\{H_{l_1}, H_{l_2}\} = 0$, the product with $H_{l_1} H_{l_2}$ at some positions can cancel out the terms with $H_{l_2} H_{l_1}$ at the same positions, for example as

$$\alpha_{l_1} \alpha_{l_2} \ldots \alpha_{l_m} (H_{l_1} H_{l_2} \ldots H_{l_m} + H_{l_2} H_{l_1} \ldots H_{l_m}) = 0. \quad (12)$$

After a given procedure of cancellation, part of L^m terms in H^m can be cancelled out via this anti-commutative relation and we rewrite H^m as the sum of remaining sequences,

$$H^m = \sum_{r=1}^{R} \alpha_{l_1^r} \alpha_{l_2^r} \ldots \alpha_{l_m^r} H_{l_1^r} H_{l_2^r} \ldots H_{l_m^r}. \quad (13)$$

where $l_1^r, \ldots, l_m^r \in \{1, \ldots, L\}$ and R is the number of remaining terms. For different methods of cancellation, we could have different R. For example, if we use the cancellation pair by pair as shown in the following, for a pair-wisely anti-commutative Hamiltonian, $R = L^{\lfloor \frac{m+1}{2} \rfloor}$. In order to estimate the norm of H^m, we mainly focus on the sum of the coefficients in the remaining R terms. Thus we define a key parameter $\alpha^{(m)}$ as

$$
\begin{aligned}
\alpha^{(m)} &= \min \sum_{r=1}^{R} \alpha_{l_1^r} \alpha_{l_2^r} \ldots \alpha_{l_m^r} \\
\text{s.t. } H^m &= \sum_{r=1}^{R} \alpha_{l_1^r} \alpha_{l_2^r} \ldots \alpha_{l_m^r} H_{l_1^r} H_{l_2^r} \ldots H_{l_m^r},
\end{aligned}
\tag{14}
$$

with an upper bound of $\|H^m\|$ as $\|H^m\| \leq \alpha^{(m)}$. Here the minimization is over all the possible cancellations.

In practice, it is generally hard to directly estimate $\alpha^{(m)}$ for a general Hamiltonian. Nevertheless, we can repetitively apply a low order cancellation to serve as an upper bound for higher order cancellations,

$$
\alpha^{(km)} \leq (\alpha^{(k)})^m.
\tag{15}
$$

Here we take a second order cancellation as an example to show the cancellation procedure. We first focus on the positions $(1, 2)$ in a sequence of H_l, and cancel out all the anti-commutative pairs on positions $(1, 2)$,

$$
\begin{aligned}
H_{l_1} H_{l_2} \ldots H_{l_m} + H_{l_2} H_{l_1} \ldots H_{l_m} &= 0, \\
\{H_{l_1}, H_{l_2}\} &= 0.
\end{aligned}
\tag{16}
$$

Then all the rest sequences satisfy

$$
H_{l_1} H_{l_2} \ldots H_{l_m}, \ (l_1, l_2) \in Comm,
\tag{17}
$$

where $Comm = \{(i, j) | [H_i, H_j] = 0\}$ denotes the set of commutative pairs and $|Comm|$ denotes the number of elements in this set. We then repetitively cancel out the anti-commutative pairs on the positions $(3, 4), \ldots, (2P-1, 2P)$ one by one, where $P = \lfloor \frac{m}{2} \rfloor$. At last, the remaining terms satisfy that

$$
\begin{aligned}
&Even \ m : H_{l_1} H_{l_2} \ldots H_{l_{2P-1}} H_{l_{2P}}, \\
&Odd \ m : H_{l_1} H_{l_2} \ldots H_{l_{2P-1}} H_{l_{2P}} H_{l_{2P+1}}, \\
&(l_{2j-1}, l_{2j}) \in Comm, \ j = 1, \ldots, P.
\end{aligned}
\tag{18}
$$

Consequently, there are $|Comm|^{\lfloor \frac{m}{2} \rfloor}$, $|Comm|^{\lfloor \frac{m}{2} \rfloor} L$ terms left in H^m for even m

and odd m, and H^m can be expressed as

$$
H^m = \sum_{(l_{2j-1}, l_{2j}) \in Comm} \alpha_{l_1} \ldots \alpha_{l_m} H_{l_1} \ldots H_{l_m},
\tag{19}
$$

Based on the above cancellation, we obtain the upper bounds for $\alpha^{(m)}$

$$
\begin{aligned}
\alpha^{(2)} &\leq \alpha_{comm}, \\
\alpha^{(m)} &\leq \alpha_{comm}^{\lfloor \frac{m}{2} \rfloor}, \ even \ m, \\
\alpha^{(m)} &\leq \alpha \alpha_{comm}^{\lfloor \frac{m}{2} \rfloor}, \ odd \ m,
\end{aligned}
\tag{20}
$$

where $\alpha_{comm} = \sum_{(i,j) \in Comm} \alpha_i \alpha_j$, and $\alpha = \sum_{l=1}^{L} \alpha_l$ denote the summation of the coefficients of commutative pairs and the summation of coefficients of all L terms, respectively. As a result, the spectral norm of H^m is bounded as

$$
\begin{aligned}
\|H^m\| &\leq \alpha_{comm}^{\lfloor \frac{m}{2} \rfloor}, \ even \ m, \\
\|H^m\| &\leq \alpha \alpha_{comm}^{\lfloor \frac{m}{2} \rfloor}, \ odd \ m.
\end{aligned}
\tag{21}
$$

Similarly, we can directly estimate the third order $\alpha^{(3)}$ and the fourth order cancellation $\alpha^{(4)}$ numerically and use them to bound $\alpha^{(m)}$.

4 Applications of anti-commutative cancellation

In this section, we show various applications of anti-commutative cancellation. We first use it to reduce the algorithmic error in the truncated Taylor series method in Sec.4.1. Secondly, we propose a modified LCU algorithm to further reduce the high order error in the truncated Taylor expansion. Finally, we explore an extreme case where all the terms are pair-wisely anti-commutative and propose a modified LCU method tailored for this Hamiltonian. Note that the first two applications are suitable for all Hamiltonians, though different Hamiltonians would have different degree of improvement over the original method.

4.1 Tightening errors in truncated Taylor series algorithm

In this section, we show how the above anti-commutative cancellation could be applied to have a tighter estimation of the algorithmic error in the Taylor series method. In the original Taylor series method with expansion up to the

K-th order, the algorithmic error comes from all the high-order terms [8, 17], which is

$$\left\| \widetilde{U}(t) - \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!} \right\| = \left\| \sum_{k=K+1}^{\infty} \frac{(-itH)^k}{k!} \right\|$$
$$\leq \sum_{k=K+1}^{\infty} \frac{(t\sum_l \alpha_l)^k}{k!}$$
$$\leq \frac{(t\sum_l \alpha_l)^{K+1}}{(K+1)!} \left[\sum_{j=0}^{\infty} \frac{(t\sum_l \alpha_l)^j}{j!} \right]$$
$$= \frac{(t\alpha)^{K+1}}{(K+1)!} e^{t\alpha}.$$

$$(22)$$

Here we take all the high-order terms into account and the inequality is not tight since some high-order terms may cancel out each other according to the above analysis.

Now, we give a refined error estimation for the Taylor series method using the above canceling method. We divide the high order terms into two parts: odd order terms and even order terms. And the spectral norm of each part can be bound via Eq. (21). We denote $\frac{\alpha}{\sqrt{\alpha_{comm}}} = q$ and the refined error can be expressed as

$$\left\| \widetilde{U}(t) - \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!} \right\|$$
$$\leq \frac{(t\alpha)^{K+1}}{(K+1)!} \frac{(q+1)e^{t\alpha/q} + (-1)^K(q-1)e^{-t\alpha/q}}{2q^{K+1}}.$$

$$(23)$$

The key parameter $q = \frac{\alpha}{\sqrt{\alpha_{comm}}}$ is related to the amount of anti-commutative relations in H. This is because $\alpha^2 = \alpha_{comm} + \alpha_{anti}$ where $\alpha_{anti} = \sum_{i,j \in Anti} \alpha_i \alpha_j$ with $Anti = \{(i,j)|\{H_i, H_j\} = 0\}$, and $1/q^2 = \alpha_{comm}/\alpha^2$ thus measures the portion of commutative pairs. Comparing to the errors in Eq. (22), we reduce it by a factor of $O(\frac{1}{q^{K+1}})$. If all the terms are commutative with $q = 1$, it recovers the previous result. However, when H is very anti-commutative with a small $1/q$ or a large q, the refined error becomes much tighter. For example, consider a Hamiltonian H with the similar magnitudes of coefficients, and half of the pairs of terms are commutative, then q is roughly $\sqrt{2}$ and we can reduce the error to $2^{-(K+1)/2}$. For a Hamiltonian H with pair-wisely anti-commutative properties, we have $\alpha_{comm} = \sum_l \alpha_l^2$, $q = \sqrt{L}$, and the error can be reduced to $L^{-K/2}$. We give more details about the refined error with third and general p order

cancellation in Appendix B.

4.2 Modified Taylor series method reducing high order truncated errors

Here in this section, we further present a refined Taylor series method that exploits anti-commutative cancellation and takes account of high order terms with negligible additional cost. We denote all the high order terms (more than $K - 1$) as

$$R_K = \frac{(-itH)^K}{K!} + \sum_{k=K+1}^{\infty} \frac{(-itH)^k}{k!}$$
$$= \frac{(-itH)^{K-1}}{K!} \left(-itH + \sum_{j=2}^{\infty} \frac{(-itH)^j}{(K+j-1)!/K!} \right)$$
$$= \frac{(-itH)^{K-1}}{K!} \left[\sum_{l=1}^{L} \alpha_l(-itH_l) + \sum_{\vec{e}} \widetilde{\gamma_{\vec{e}}} H_{\vec{e}} \right].$$

$$(24)$$

Here in the last line, we rewrite the terms $-iHt$ and $\sum_{j=2}^{\infty} \frac{(-itH)^j}{(K+j-1)!/K!}$ as $\sum_{l=1}^{L} \alpha_l(-itH_l)$, $\sum_{\vec{e}} \widetilde{\gamma_{\vec{e}}} H_{\vec{e}}$, respectively. Here $H_{\vec{e}}$ is the product of H_l that remains after a certain canceling procedure which can be expressed as

$$H_{\vec{e}} = (-i)^k H_{e_1} H_{e_2} \dots H_{e_k}. \quad (25)$$

In the original truncated Taylor LCU method, all these $H_{\vec{e}}$ are discarded as error terms. However, some $H_{\vec{e}}$ may equal to simple unitaries, such as identity I, or unitaries that appears in lower order expansions, such as H_l. For example, according to the anti-commutative cancellation, the second and third order terms H^2 and H^3 can be expressed as

$$H^2 = \beta_0 I + \sum_{(l_1, l_2) \in Comm^*} \alpha_{l_1} \alpha_{l_2} H_{l_1} H_{l_2},$$
$$H^3 = \sum_l \beta_l H_l +$$
$$\sum_{(l_1, l_2),(l_1, l_3),(l_2, l_3) \in Comm^*} \alpha_{l_1} \alpha_{l_2} \alpha_{l_3} H_{l_1} H_{l_2} H_{l_3},$$

$$(26)$$

where we define the set containing all the commutative pairs without the same terms as $Comm^* = \{(i,j)|[H_i, H_j] = 0, \ i \neq j\}$. Here we use the facts $H_l^2 = I$ and $H_l H_{l'} H_{l'} = H_l$. For high order terms, the following sequences equal to I and H_l,

$$Even \ length : (H_{l_1} H_{l_1}) \dots (H_{l_P} H_{l_P}) = I,$$
$$Odd \ length : (H_{l_1} H_{l_1}) \dots (H_{l_P} H_{l_P}) H_l = H_l.$$

$$(27)$$

Instead of discarding all these terms, we can update the coefficients of $-iH_l$ from $\alpha_l t$ to a new one γ_l that includes higher order contributions and add the term $U_0 = -I$. Moreover, we can add other unitaries with the modified coefficients apart from the unitaries $-iH_l$ and $U_0 = -I$. For example, we choose E number of unitaries with the largest coefficients in the rest of $H_{\vec{e}}$, denoted as U_1 to U_E. Consequently, we construct an approximate K-th order term as

$$\widetilde{R}_K = \frac{(-itH)^{K-1}}{K!}\left[\sum_{l=1}^L \gamma_l(-iH_l) + \sum_{j=0}^E \widetilde{\gamma_j}U_j\right], \tag{28}$$

and we can implement

$$\widetilde{U}(t) = \sum_{k=0}^{K-1}\frac{(-itH)^k}{k!} + \widetilde{R}_K \tag{29}$$

via the LCU routine. Specifically, we implement the low order (less than $K-1$) terms as usual and the K-th term $\widetilde{R}_K$ with a modified circuit that involves little additional resource. The details of additional gates are shown in the numerical section. Overall, the refined method could reduce the error with only a few additional gates. It works well for Hamiltonians with large anti-commutative terms or very biased coefficients.

Now we analyze the error for the modified scheme. We divide the error into three parts $(K+1)$-th order error E_{K+1}, $(K+2)$-th order error E_{K+2}, and high order error E_∞ as

$$\left\|\widetilde{U}(t) - \sum_{k=0}^\infty \frac{(-itH)^k}{k!}\right\| = \|E_{K+1} + E_{K+2} + E_\infty\|. \tag{30}$$

Without loss of generality, we assume K is odd. Benefiting from anti-commutative relation, we could obtain a more accurate analysis of E_{K+1} and E_{K+2} which take up a large proportion of the total error and are bound as

$$\begin{aligned}\|E_{K+1}\| &\leq \frac{t^{K+1}\alpha_{comm}^{\frac{K-1}{2}}E_\epsilon}{(K+1)!}, \\ \|E_{K+2}\| &\leq \frac{t^{K+2}\alpha_{comm}^{\frac{K-1}{2}}\alpha_r^{(3)}}{(K+2)!},\end{aligned} \tag{31}$$

where E_ϵ is the sum of coefficients of the remaining terms in $\sum_{(l_1,l_2)\in Comm^*}\alpha_{l_1}\alpha_{l_2}H_{l_1}H_{l_2}$ in Eq. (26), which are not included in U_j and $-iH_l$ and $E_\epsilon \leq \alpha_{comm} - (\sum_l \alpha_l^2)$, and $\alpha_r^{(3)}$ is the sum of

coefficients $\alpha_{l_1}\alpha_{l_2}\alpha_{l_3}$ in H^3 according to Eq. (26). We could directly use the bound in Eq. (23) for high order terms $\|E_\infty\|$. Though we could also calculate the sequences which equals to H_l or I in E_∞, it would not influence the results a lot.

With the parameter $\frac{\alpha}{\sqrt{\alpha_{comm}}} = q$, the error can be expressed as

$$\begin{aligned}&\left\|\widetilde{U}(t) - \sum_{k=0}^\infty \frac{(-itH)^k}{k!}\right\| \\ &\leq \frac{(t\alpha)^{K+1}}{(K+1)!q^{K-1}}\frac{E_\epsilon}{\alpha^2} + \frac{(t\alpha)^{K+2}}{(K+2)!q^{K-1}}\frac{\alpha_r^{(3)}}{\alpha^3} \\ &+ \frac{(t\alpha)^{K+3}}{(K+3)!}\frac{(q+1)e^{t\alpha/q} + (-1)^K(q-1)e^{-t\alpha/q}}{2q^{K+3}}.\end{aligned} \tag{32}$$

We refer to Appendix A and C for details of the implementation and error analysis. From the above expression, we know that this error is related to some key parameters α_{comm}, E_ϵ and $\alpha_r^{(3)}$. Though this modified method could not influence asymptotic performance, it could significantly reduce the error in the near term implementation of the Taylor series method. The numerical verification of the improvement on performance is shown in the next section.

4.3 Modified Taylor series method with pair-wisely anti-commutative terms

At last, we consider an extreme case with Hamiltonians consisting of pair-wisely anti-commutative terms. For example, we construct an n-qubit Hamiltonian $H = \sum_{j=1}^n H_j$ with n pair-wisely anti-commutative terms,

$$\begin{aligned}&H_1 = X_1, H_2 = Z_1 \otimes Z_2 \\ &H_j = Z_1 \otimes X_2 \cdots \otimes X_{j-1} \otimes Z_j, j = 3,\ldots,n.\end{aligned} \tag{33}$$

In this section, we show how the evolution of this kind of Hamiltonians can be simulated via the LCU method without any approximation error. We first show the exact form of the expansion of the Hamiltonian.

Lemma 1. *For a Hamiltonian $H = \sum_{l=1}^L \alpha_l H_l$ where all the terms H_l are unitary and mutually anti-commutative, $\{H_{l_1}, H_{l_2}\} = 0$, $l_1 \neq l_2$, H^m can be expressed as*

$$H^m = \gamma_0^{(m)}I + \sum_{l=1}^L \gamma_l^{(m)}H_l, \tag{34}$$

for some coefficients $\gamma_l^{(m)}$, $l = 0,\ldots,L$.

In particular, we have

$$\gamma_0^{(2j-1)} = 0, \ \gamma_l^{(2j-1)} = \beta_s^{2j-2}\alpha_l,$$
$$\gamma_0^{(2j)} = \beta_s^{2j}, \ \gamma_l^{(2j)} = 0, \tag{35}$$

where $j \in \mathbb{N}^+$ is a positive integer and $\beta_s = \sqrt{\sum_{l=1}^{L} \alpha_l^2}$. Now, we expand the evolution $U(t) = e^{-iHt}$, and we can rewrite it as

$$U(t) = \widetilde{U}(t) = \tilde{\alpha}_0 I + \sum_l \tilde{\alpha}_l(-iH_l), \tag{36}$$

where

$$\tilde{\alpha}_0 = 1 - \beta_s^2 t^2/2! + \beta_s^4 t^4/4! + \cdots = \cos(t\beta_s),$$
$$\tilde{\alpha}_l = \alpha_l t - \alpha_l \beta_s^2 t^3/3! + \alpha_l \beta_s^4 t^5/5! + \dots$$
$$= \frac{\alpha_l}{\beta_s} \sin(t\beta_s), \ l = 1, \dots, L. \tag{37}$$

Note that here the coefficients $\tilde{\alpha}_l$ may not be always positive, and we could replace them with the absolute values and change the sign of the unitary gate accordingly. The sum of the absolute value coefficients is

$$s = \sum_l |\tilde{\alpha}_l| = |\cos(t\beta_s)| + \frac{\alpha}{\beta_s}|\sin(t\beta_s)|, \tag{38}$$

which is a function of t and β_s. In particular, we have $L\beta_s^2 \geq \alpha^2$ and thus $1 \leq s \leq 1 + \sqrt{L}$.

Theorem 1. *For a Hamiltonian $H = \sum_{l=1}^{L} \alpha_l H_l$ acting on n qubits where all the terms H_l are mutually anti-commutative, $\{H_{l_1}, H_{l_2}\} = 0$, $l_1 \neq l_2$, we can implement the evolution $U(t) = e^{-iHt}$ without algorithmic error via the LCU method with the gate complexity $\mathcal{O}(L^{\frac{3}{2}}(n + \log L))$.*

Here the gate complexity $\mathcal{O}(L^{\frac{3}{2}}(n + \log L))$ is independent of t and ε. The details of the implementation of LCU is shown in Appendix D.2. Moreover, the expansion of the unitary also indicates efficient classical simulation of the Hamiltonian similar to Hamiltonians with mutually commutative terms. We can also extend the above strict case to more general cases.

Corollary 1. *Suppose there exists a positive integer M, satisfying that $H^M = \gamma I$, we can implement the evolution $U(t) = e^{-iHt}$ without algorithmic error via the LCU method.*

Similarly, we rewrite the unitary as

$$U(t) = \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!} = \sum_{k=0}^{M-1} \sum_{j=0}^{\infty} (-itH)^k \frac{(-it)^{jM}\gamma^j}{(k+jM)!},$$
$$= \sum_{k=0}^{M-1} \gamma_k(-itH)^k, \tag{39}$$

where coefficients $\gamma_k = \sum_{j=0}^{\infty} \frac{(-it)^{jM}\gamma^j}{(k+jM)!}$. We can implement this unitary like the original truncated Taylor method with at most $M-1$ order without any algorithmic error. In the above analysis, we require that the Hamiltonians are mutually anti-commutative, which is rather strict.

Corollary 2. *When the Hamiltonian H is not perfectly anti-commutative, but close to the perfect case,*

$$H^2 = \beta_s^2 I + V, \quad \|V\| \leq \varepsilon_A. \tag{40}$$

There exist coefficients $\tilde{\alpha}_l$, such that $\widetilde{U}(t)$ is close to $U_0(t) = \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!}$ with an error $\mathcal{O}(t^2\varepsilon_A) + \mathcal{O}(t^3\varepsilon_A\alpha)$ for a small t,

$$\widetilde{U}(t) = \tilde{\alpha}_0 I + \sum_l \tilde{\alpha}_l(-iH_l),$$
$$\left\|\widetilde{U}(t) - \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!}\right\| = \mathcal{O}(t^2\varepsilon_A) + \mathcal{O}(t^3\varepsilon_A\alpha). \tag{41}$$

Note that here we need a small time t. Thus for a long time t, we could also divide it into r segments, and the total error is $\mathcal{O}(t^2\varepsilon_A/r) + \mathcal{O}(t^3\varepsilon_A\alpha/r^2)$. When $t = n$, and $\alpha = \mathcal{O}(n)$, then the error is $\mathcal{O}(n^2\varepsilon_A/r)$. The parameter for commutative pairs has the upper bound, $\varepsilon_A \leq \sum_{(i,j)\in Comm^*} \alpha_i\alpha_j < \alpha_{comm}$. Compared to the error in the first order PF $\mathcal{O}(n^2\alpha_{anti}/r)$, the differences are the prefactors quantifying the anti-commutation and commutation property. When the Hamiltonian is more anti-commutative, $\alpha_{comm} < \alpha_{anti}$, then this approximate LCU method could has the advantage over the first order PF.

5 Numerical simulation

Here in this section, we show the numerical performance of the proposed methods. Lattice Hamiltonians mostly consist of low weight terms, which are likely to be commutative to each

other; whereas chemistry Hamiltonians, after the fermion-to-qubit encoding, have more anti-commutative terms. Since the proposed methods are more suitable for an anti-commutative Hamiltonian, we focus on the electronic structure of chemistry Hamiltonians. The second quantized form of the electronic Hamiltonians is

$$H = \sum_{p,q} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s, \quad (42)$$

where $a^\dagger$ and a are creation and annihilation operators [27]. Then it is converted to a qubit Hamiltonian via the Jordan-Wigner transformation [22].

$$
\begin{aligned}
a_p &= \frac{X_p + iY_p}{2} \otimes Z_{p-1} \otimes \cdots \otimes Z_0, \\
a_p^\dagger &= \frac{X_p - iY_p}{2} \otimes Z_{p-1} \otimes \cdots \otimes Z_0.
\end{aligned}
\quad (43)
$$

We consider 23 different molecules as shown in Table 1 and the Hamiltonian data is from Ref. [28].

5.1 Refined errors estimation

In the Taylor series method, the total time is divided into several segments with each time segment $t = \ln 2/\alpha$. The error in the original analysis is

$$\varepsilon_o = \frac{\delta_o^2 + 3\delta_o + 4}{2} \delta_o, \ \delta_o = 2\frac{(\ln 2)^{K+1}}{(K+1)!}. \quad (44)$$

where δ_o is the truncated error in the Taylor expansion. According to Eq. (23), we calculate the refined errors ε_n and show the improvement of the refined error estimation for various molecules with up to 25 qubits. In order to show the improvement clearly, we calculate the ratios $\varepsilon_o/\varepsilon_n$ versus different truncated order $K = 10, 20, 30, 40$ in Table 1 using the second order cancellation. Moreover, we also plot the ratio $\varepsilon_o/\varepsilon_n$ for some example molecules, HO, LiH, BH, and BeH_2 using second order cancellation, third order, and fourth order cancellation in Fig. 1 using the upper bounds in Eq. (61). The lower bounds of the key parameters $q = \frac{\alpha}{(\alpha^{(2)})^{1/2}}, \frac{\alpha}{(\alpha^{(3)})^{1/3}}, \frac{\alpha}{(\alpha^{(4)})^{1/4}}$ can be estimated numerically, which are summarized in Table 2. Though higher order cancellation brings more significant improvement, the numerical estimation of high order cancellation also becomes much harder with the increase of the number of terms L and the number of qubits n.

			Error Reduction $\varepsilon_o/\varepsilon_n$			
Formula	Qubits	L	K = 10	K = 20	K = 30	K = 40
HO	11	631	1.445	2.016	2.813	3.926
LiH	11	631	1.866	3.285	5.782	10.177
BH	11	631	1.962	3.615	6.660	12.270
BeH_2	13	666	1.990	3.714	6.930	12.933
NH_2	13	1086	1.655	2.611	4.119	6.500
BH_2	13	1086	2.111	4.157	8.187	16.123
CH_3	15	1969	1.919	3.466	6.260	11.305
NH_3	15	2929	1.806	3.085	5.272	9.007
CH_4	17	6892	2.492	5.708	13.074	29.948
NO	19	4427	1.428	1.972	2.724	3.761
CN	19	5851	1.561	2.337	3.498	5.237
BN	19	5851	1.652	2.604	4.105	6.469
LiOH	21	8750	1.639	2.565	4.013	6.278
HBO	21	8758	1.663	2.637	4.180	6.627
HOF	21	12070	1.525	2.234	3.273	4.796
CHF	21	12074	1.616	2.497	3.857	5.958
H_2NO	23	9257	1.532	2.254	3.317	4.881
CH_2O	23	9257	1.582	2.396	3.630	5.499
NH_2F	23	15673	1.574	2.374	3.581	5.401
CH_2F	23	15681	1.605	2.465	3.785	5.811
CH_3F	25	18600	1.575	2.375	3.583	5.405
CH_3Li	25	19548	2.014	3.799	7.166	13.518
OCH_3	25	39392	1.705	2.763	4.480	7.263

Table 1: Error Reduction using second-order cancellation $\varepsilon_o/\varepsilon_n$ for various molecules.

Molecule	$\frac{\alpha}{(\alpha^{(2)})^{1/2}}$	$\frac{\alpha}{(\alpha^{(3)})^{1/3}}$	$\frac{\alpha}{(\alpha^{(4)})^{1/4}}$
HO	1.0339	1.0552	1.0699
LiH	1.0582	1.0949	1.1177
BH	1.0630	1.0994	1.1222
BeH_2	1.0644	1.1061	1.1324

Table 2: Parameters $\frac{\alpha}{(\alpha^{(2)})^{1/2}}, \frac{\alpha}{(\alpha^{(3)})^{1/3}}, \frac{\alpha}{(\alpha^{(4)})^{1/4}}$ estimation for second order, third order, and fourth order cancellation.

5.2 Reducing the high order error in modified algorithm

In Section 4.2, we further reduce the high order truncated error via modifying the coefficients and involving more unitary operations in the Taylor series. We calculate three different types of new errors with $t = \ln 2/\alpha$:

1. Refined errors shown in Eq. (23) using second order cancellation;

2. Reduced errors in modified method without additional unitaries ($E = 0$) in Eq. (32).

3. Reduced errors in modified method with $E = 2^w - L - 1$ additional unitaries in Eq. (32) where $w = \lceil \log_2 L \rceil$.
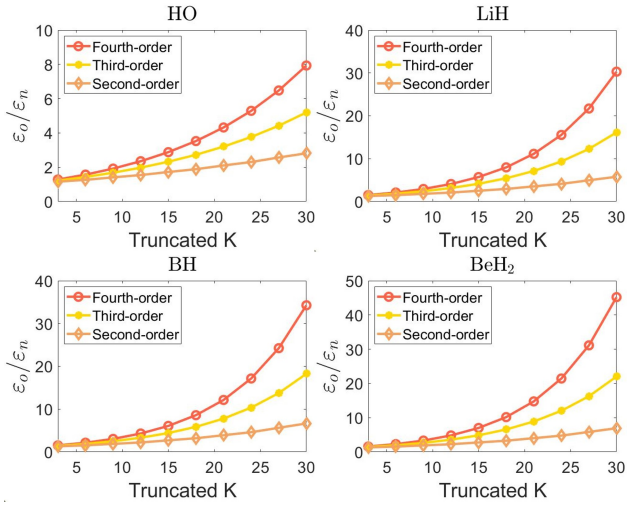
Figure 1: Ratio $\varepsilon_o/\varepsilon_n$ with second order, third order, and fourth order cancellation versus different truncated order K with $t = \ln 2/\alpha$ for molecules HO, LiH, BH, BeH_2.

Figure 2: Ratio $\varepsilon_o/\varepsilon_n$ versus different truncated order K with $t = \ln 2/\alpha$ for molecules BH, CH_4, HBO, CH_3Li. Modified LCU 1 and 2 denote the results with $E = 0$ and $E = 2^w - L - 1$ additional unitaries in Eq. (32), respectively.

According to Appendix C and Ref. [17], these three schemes we applied in the numerical simulation have almost the same gate cost. In order to show the improvement clearly, we show the curves of the ratio of the original error to new calculated error, $\varepsilon_o/\varepsilon_n$, versus different truncated order K for molecules, BH, CH_4, HBO, CH_3Li with terms L ranging from 631 to 19548. We find that with increasing K, the reduced error becomes more significant. Note that here the improvement is not related to the system size n nor the number of terms L, but related to the anti-commutative property of different molecules.

In some Hamiltonian simulation tasks, a predetermined simulation accuracy ε is given. The task is to find the most efficient circuit approximating the ideal evolution with at most ε error. Thus, we also compare the required truncated K with a given simulation accuracy ε for molecules BH, CH_4, HBO, CH_3Li. We choose the simulation time $t = n$ where n is the number of qubits. We find that in most of cases shown in Table 3, the refined error scheme or the modified scheme could reduce one order of Taylor expansion. Although this does not seem a significant change in theory, it can indeed reduce both the gate count and the number of ancillary qubits, making its practical implementation much easier. For a Hamiltonian with L unitaries, one order reduction could reduce $\log L$ number of ancillary qubits. We show the total CNOT gates and ancillary qubits reduction for these examples in Table 4. For example, consider a large molecule Hamil-
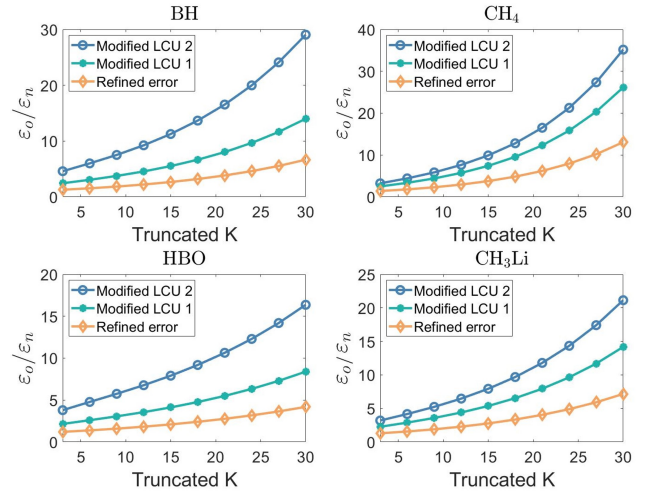
tonian with $N = 100$ spin orbitals and $h = 50$ electrons, we have about $L \approx N^2 h^2 = 2.5 \times 10^7$ terms and in one segment our method can roughly reduce 25 qubits and accordingly remove 7.5×10^8 CNOT gates that act on those qubits.

6 Discussion

In this work, we explore the anti-commutative relation in Hamiltonian simulation. We find that Anti-commutative relation plays a positive role in the truncated Taylor series method. We show that the simulation accuracy could be significantly improved by utilizing the anti-commutative relation. In particular, pair-wisely anti-commutative Hamiltonians admit an efficient and error free simulation with the LCU method. Our work is the first attempt of exploiting anti-commutative relations in Hamiltonian reduction, which may inspire and motivate more interesting and significant future works to find more efficient reduction mechanisms that combine with other techniques or considering specific Hamiltonians that have large anti-commutative terms.

When we consider some specific types of Hamiltonians, the performance of Taylor series method could be further improved. For instance, combining our result with techniques in Ref. [29], we may further reduce the simulation error for electronic structure Hamiltonians. Here most of our results are based on a low order anti-

Error ε	Minimum K											
	BH			CH$_4$			HBO			CH$_3$Li		
1e-06	10	10	10	11	11	10	11	11	11	11	11	10
1e-07	11	11	10	12	11	11	12	12	11	12	12	11
1e-08	12	12	11	12	12	12	13	13	12	13	12	12
1e-09	13	13	12	13	13	13	14	13	13	13	13	13
1e-10	14	13	13	14	14	13	14	14	14	14	14	14
1e-11	14	14	14	15	14	14	15	15	14	15	15	14
1e-12	15	15	14	15	15	15	16	16	15	16	15	15
1e-13	16	15	15	16	16	15	16	16	16	16	16	16
1e-14	16	16	16	17	16	16	17	17	17	17	17	16
1e-15	17	17	16	18	17	17	18	18	17	18	17	17
1e-16	18	18	17	18	18	18	19	18	18	18	18	18
1e-17	19	18	18	19	19	18	19	19	19	19	19	18
1e-18	19	19	18	20	19	19	20	20	19	20	19	19
1e-19	20	20	19	20	20	20	21	20	20	21	20	20
1e-20	21	20	20	21	21	20	21	21	21	21	21	20

Table 3: Minimum K for required accuracy in three different schemes for molecules BH, CH$_4$, HBO, CH$_3$Li. The first column for each molecule is with original Taylor series method. The second column is with refined error scheme in Sec. 4.1. And the third column is with modified scheme in Sec. 4.2 with $E = 2^w - L - 1$.

Reduction	BH	CH$_4$	HBO	CH$_3$Li.
CNOT Gate	8.7×10^6	3.0×10^8	1.5×10^9	2.3×10^9
Ancillary qubits	10	13	14	15

Table 4: Number of CNOT gates and ancillary qubits reduction for one order reduction of Taylor expansion for simulating molecules BH, CH$_4$, HBO, CH$_3$Li with simulation time $t = n$ where n is the number of qubits.

commutative cancellation. In general, high order anti-commutative cancellation is classically intractable. However, we may find explicit and more efficient strategies to exploit the high order anti-commutative cancellation for some specific Hamiltonians.

Another direction is exploring the role of anti-commutation in other Hamiltonian simulation algorithms, e.g. product formula methods and quantum signal processing algorithm. From our current understanding, it might seem unlikely that anti-commutative relation could help in a standard product formula methods due to the lack of anti-commutation cancellation. But it might be helpful for the variant of product formula methods, e.g. randomized product formula,

which involves the summation of the products of different sequences [13, 14, 18]. Meanwhile, due to the complementary advantages of commutation in product formula and anti-commutation in the Taylor series method, we could also consider hybrid approaches of Trotter and Taylor series methods. In this case, the Hamiltonian could be divided into commutative and anti-commutative terms (sets), which are simulated using Trotter and Taylor expansion respectively. Of course, how to divide the terms and whether such a hybrid approach is useful seem a very interesting future direction.

The error reduction and speed-up in this work mainly come from representing the power of Hamiltonian H with the summation of fewer unitary terms. Thus, the cancellation idea is not restricted to anti-commutative cancellation. For instance, if the Hamiltonian H has certain symmetry properties, then any order of H could be expressed in the summation of the terms in the symmetry subspace, which might be useful for designing more efficient Hamiltonian simulation algorithms.

Note that the perfect anti-commutation case admits an efficient classical simulation of Hamiltonian evolution. And the Trotter formula, for instance the second order trotter method is used in the classical algorithms of estimating the partition function [9, 10]. We expect the anti-commutation property may inspire better classical simulation method of quantum problems.

7 Acknowledgement

A Taylor expansion method

In order to implement $\widetilde{U}(t)$, we need three key elements: a state preparation procedure accomplished by oracle G, a reflection denoted as R, and an oracle selecting unitary operations denoted as select(V).

We assume that the ancillary system is a d dimensional Hilbert space $\mathcal{H}_{an}$ and the evolution operation e^{-iHt} acts on an n qubit Hilbert space $\mathcal{H}_s$ with dimension 2^n.

Here the oracle G is a unitary that transforms the ancillary state $|0\rangle$ to

$$G|0\rangle = \frac{1}{\sqrt{s}} \sum_j \sqrt{\beta_j} |j\rangle, \tag{45}$$

where s is the normalized parameter,

$$s = \sum_{j=0}^{J} \beta_j = \sum_{k=0}^{K} \frac{(t \sum_l a_l)^k}{k!}. \tag{46}$$

The select(V) is defined as

$$\text{select}(V) = \sum_j |j\rangle\langle j| \otimes V_j. \tag{47}$$

The control register $|j\rangle\langle j|$ acts on the ancillary system $\mathcal{H}_{an}$. We construct an operator as

$$W = (G^\dagger \otimes I)\text{select}(V)(G \otimes I), \tag{48}$$

which when applied to the input state $|0\rangle \otimes |\Psi\rangle$ gives

$$W(|0\rangle \otimes |\Psi\rangle) = \frac{1}{s}(|0\rangle \otimes \widetilde{U}(t)|\Psi\rangle) + \frac{\sqrt{s^2-1}}{s}|\phi\rangle, \tag{49}$$

where $|\phi\rangle$ is orthogonal to the subspace $span\{|0\rangle \otimes \mathcal{H}_s\}$. After projecting to $|0\rangle$ of the ancillary system $\mathcal{H}_{an}$, we can obtain the approximate evolution $\widetilde{U}(t)|\Psi\rangle$ with probability $1/s$. However, the failure probability is nonignorable, thus we need to boost the coefficient before $|0\rangle \otimes U(t)|\Psi\rangle$. When $s = 2$, we could use the oblivious amplitude amplification to boost the probability,

$$-WRW^\dagger RW(|0\rangle \otimes |\Psi\rangle) = |0\rangle \otimes \widetilde{U}(t)|\Psi\rangle, \tag{50}$$

where R is the reflection $R = (I - 2|0\rangle\langle 0|) \otimes I$ acting on $\mathcal{H}_{an} \otimes \mathcal{H}_s$. Here oblivious amplitude amplification is similar to the iterations in the Grover algorithm. The operation RW can be understood as the reflection along the axis vector $|0\rangle \otimes U(t)|\Psi\rangle$. After this, we implement $-WRW^\dagger = 2W|0\rangle\langle 0|W^\dagger - I$, which is the reverse reflection along the axis $W(|0\rangle \otimes |\Psi\rangle)$ and completes the amplification. Thus the post selected evolution $\widetilde{V}(t)$ is

$$\widetilde{V}(t) = (\langle 0| \otimes I) - WRW^\dagger RW(|0\rangle \otimes I). \tag{51}$$

In general, s can be far away from 2. Similar to the product formula method, here we could divide the simulation time t into r segments such that the simulation time in each segment $\tau = t/r = \frac{\ln 2}{\sum_l \alpha_l}$ satisfies

$$s = \sum_{k=0}^{K} \frac{(t \sum_l a_l/r)^k}{k!} \approx exp\left(t \sum_l a_l/r\right) = 2. \tag{52}$$

For the remaining time τ_{re} less than τ, $\tau = \frac{\ln 2}{\sum_l \alpha_l}$, $t = (r-1)\tau + \tau_{re}$. Another ancillary qubit can be used to exactly boost the value of s to 2, see Refs [8, 17]. Thus the whole evolution can be implemented by

$$\widetilde{U}_{TS}(t) = \widetilde{V}(\tau_{re})\widetilde{V}(\tau)^{r-1}. \tag{53}$$

The algorithmic error mainly comes from the finite truncated Taylor series

$$\|\widetilde{U}(\tau) - U_0(\tau)\| \leq 2\frac{(\ln 2)^{K+1}}{(K+1)!}. \tag{54}$$

It can be verify that

$$\widetilde{V}(\tau) = \frac{3}{s}\widetilde{U}(\tau) - \frac{4}{s^3}\widetilde{U}(\tau)\widetilde{U}(\tau)^\dagger\widetilde{U}(\tau). \tag{55}$$

Supposing the truncated error $\delta = 2\frac{(\ln 2)^{K+1}}{(K+1)!}$, the distance for each segment can be bound as

$$\|\widetilde{V}(\tau) - U_0(\tau)\| \leq \frac{\delta^2 + 3\delta + 4}{2}\delta, \tag{56}$$

and the total error accumulates with r segments as

$$\left\|\widetilde{V}(\tau_{re})\widetilde{V}(\tau)^{r-1} - e^{-iHt}\right\| \leq \frac{\delta^2 + 3\delta + 4}{2}\delta r. \tag{57}$$

Given a fixed total allowed error ε, the error for each segment is $O(\varepsilon/r)$. In order to make Eq. (56) satisfy the error constraint, the parameter K can be chosen as $K = O\left(\frac{\log \frac{\alpha t}{\varepsilon}}{\log\log \frac{\alpha t}{\varepsilon}}\right)$, where $\alpha = \sum_l \alpha_l$. The segment number is $r = \frac{\alpha t}{\ln 2}$. In each segment, we need to implement G and select(V). The oracle G can be implemented by a unary encoding which needs $O(KL)$ gates. And select(V) mainly contributes to the gate complexity, which can be implemented via K controlled-select(H) requiring the gate cost $O(KL(n + \log L))$. Thus the total gate complexity is $O\left(\alpha t L(n + \log L)\frac{\log \frac{\alpha t}{\varepsilon}}{\log\log \frac{\alpha t}{\varepsilon}}\right)$. We refer to Refs [8, 17] for more details of this algorithm.

B Reducing the errors in the original LCU scheme

Here we give the refined error estimation for the Taylor series method using the second order, third order cancelling method. The fourth order cancelling is similar. In the second order cancellation, we can divide the high order terms into two parts: odd order terms and even order terms. And the spectral norm of each part can be bound via Eq. (21),

$$\begin{aligned}
&\left\|\widetilde{U}(t) - \sum_{k=0}^{\infty}\frac{(-itH)^k}{k!}\right\| \\
&\leq \sum_{j=0,1}\left\|\sum_{k=K+1,\ k=j\ (mod\ 2)}^{\infty}\frac{(-itH)^k}{k!}\right\| \\
&\leq \sum_{j=0,1}\left(\sum_{k=K+1,\ k=j\ (mod\ 2)}^{\infty}\frac{t^k \alpha_{comm}^{\lfloor\frac{k}{2}\rfloor}\alpha^j}{k!}\right) \\
&\leq \frac{(t\sqrt{\alpha_{comm}})^{K+1}}{(K+1)!}\sum_{j=0,1}\left(\sum_{l\geq 0,\ K+1+l=j\ (mod\ 2)}q^j\frac{t^l}{l!}\sqrt{\alpha_{comm}}^l\right) \\
&= \frac{(t\sqrt{\alpha_{comm}})^{K+1}}{2(K+1)!}\left[q\left(e^{t\sqrt{\alpha_{comm}}} + (-1)^K e^{-t\sqrt{\alpha_{comm}}}\right) + \left(e^{t\sqrt{\alpha_{comm}}} + (-1)^{K-1}e^{-t\sqrt{\alpha_{comm}}}\right)\right] \\
&= \frac{(t\alpha)^{K+1}}{(K+1)!}\frac{(q+1)e^{t\alpha/q} + (-1)^K(q-1)e^{-t\alpha/q}}{2q^{K+1}}.
\end{aligned} \tag{58}$$

Here we give the explicit form of the error via third order cancellation. For simplicity, we suppose $K = 2\ (mod\ 3)$ and denote $\frac{\alpha}{(\alpha^{(3)})^{1/3}} = q_3$ and the trivial parameters $q_0 = q_1 = 1$. Then the error are

divided into three parts as follows. The cases $K = 0$ or 1 $(mod\ 3)$ are similar.

$$
\begin{aligned}
&\left\| \widetilde{U}(t) - \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!} \right\| \\
&\leq \sum_{j=0,1,2} \left\| \sum_{k=K+1,\ k=j\ (mod\ 3)}^{\infty} \frac{(-itH)^k}{k!} \right\| \\
&\leq \sum_{j=0,1,2} \sum_{k=K+1,\ k=j\ (mod\ 3)}^{\infty} \frac{t^k (\alpha^{(3)})^{\lfloor \frac{k}{3} \rfloor} \alpha^{(j)}}{k!} \\
&\leq \frac{[t(\alpha^{(3)})^{1/3}]^{K+1}}{(K+1)!} \sum_{j=0,1,2} \left(\sum_{l=j\ (mod\ 3),\ l \geq 0} (\frac{q_3}{q_j})^j \frac{t^l}{l!} (\alpha^{(3)})^{l/3} \right) \\
&= \frac{(t\alpha)^{K+1}}{(K+1)!} \frac{S_0 + q_3 S_1 + (\frac{q_3}{q_2})^2 S_2}{q_3^{K+1}},
\end{aligned}
\tag{59}
$$

where

$$
\begin{aligned}
S_0 &= \sum_{l=0\ (mod\ 3),\ l \geq 0} \frac{t^l}{l!} (\alpha^{(3)})^{l/3} = \frac{1}{3}(e^{t\alpha/q_3} + e^{\omega t\alpha/q_3} + e^{\omega^2 t\alpha/q_3}), \\
S_1 &= \sum_{l=1\ (mod\ 3),\ l \geq 0} \frac{t^l}{l!} (\alpha^{(3)})^{l/3} = \frac{1}{3}(e^{t\alpha/q_3} + \omega^2 e^{\omega t\alpha/q_3} + \omega e^{\omega^2 t\alpha/q_3}), \\
S_2 &= \sum_{l=2\ (mod\ 3),\ l \geq 0} \frac{t^l}{l!} (\alpha^{(3)})^{l/3} = \frac{1}{3}(e^{t\alpha/q_3} + \omega e^{\omega t\alpha/q_3} + \omega^2 e^{\omega^2 t\alpha/q_3}), \\
&\omega^2 + \omega + 1 = 0.
\end{aligned}
\tag{60}
$$

For a p-th order cancellation, we denote the parameter $\frac{\alpha}{(\alpha^{(p)})^{1/p}} = q_p$ and could give a general result,

$$
\begin{aligned}
&\left\| \widetilde{U}(t) - \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!} \right\| \\
&\leq \sum_{j=0}^{p-1} \left\| \sum_{k=K+1,\ k=j\ (mod\ p)}^{\infty} \frac{(-itH)^k}{k!} \right\| \\
&\leq \sum_{j=0}^{p-1} \sum_{k=K+1,\ k=j\ (mod\ p)}^{\infty} \frac{t^k (\alpha^{(p)})^{\lfloor \frac{k}{p} \rfloor} \alpha^{(j)}}{k!} \\
&\leq \sum_{j=0}^{p-1} \sum_{k=K+1,\ k=j\ (mod\ p)}^{\infty} \frac{\alpha^{(j)}}{(\alpha^{(p)})^{j/p}} \frac{t^k (\alpha^{(p)})^{\frac{k}{p}}}{k!} \\
&\leq \frac{[t(\alpha^{(p)})^{1/p}]^{K+1}}{(K+1)!} \max_j (\frac{q_p}{q_j})^j \sum_{l \geq 0} \frac{t^l}{l!} (\alpha^{(p)})^{l/p} \\
&\leq \frac{(t\alpha)^{K+1}}{(K+1)!} \frac{e^{t\alpha/q_p}}{q_p^{K+1}} \max_j (\frac{q_p}{q_j})^j.
\end{aligned}
\tag{61}
$$

C Details of modified LCU Algorithm reducing high order error

In the modified LCU method, we construct an approximate K-th order term as

$$
\widetilde{R}_K = \frac{(-itH)^{K-1}}{K!} \left[\sum_{l=1}^{L} \gamma_l (-iH_l) + \sum_{j=0}^{E} \widetilde{\gamma_j} U_j \right].
\tag{62}
$$

According to the anti-commutative cancellation, the second and third order terms H^2 and H^3 in $\sum_{\vec{e}} \widetilde{\gamma_{\vec{e}}} H_{\vec{e}}$ are

$$H^2 = \beta_0 I + \sum_{(l_1, l_2) \in Comm^*} \alpha_{l_1} \alpha_{l_2} H_{l_1} H_{l_2},$$

$$H^3 = \sum_l \beta_l H_l + \sum_{(l_1, l_2), (l_1, l_3), (l_2, l_3) \in Comm^*} \alpha_{l_1} \alpha_{l_2} \alpha_{l_3} H_{l_1} H_{l_2} H_{l_3} \tag{63}$$

where $Comm^* = \{(i, j) | [H_i, H_j] = 0, \ i \neq j\}$. Here the first line H^2 is directly from the anti-commutative cancellation. For H^3 terms, when l_1, l_2, l_3 are equal $l_1 = l_2 = l_3$ or two of them are equal, for example, $l_1 = l_2 \neq l_3$. Then we have $H_{l_1} H_{l_2} H_{l_3} = H_{l_3}$. When l_1, l_2, l_3 are different, there are two cases: 1. All the pairs are commutative, 2. There is at least one pair H_{l_1} and H_{l_2} which are anti-commutative. For case 2, all the possible six sequences can be cancelled pair by pair as

$$H_{l_1} H_{l_2} H_{l_3} + H_{l_2} H_{l_1} H_{l_3} = 0,$$
$$H_{l_1} H_{l_3} H_{l_2} + H_{l_2} H_{l_3} H_{l_1} = 0, \tag{64}$$
$$H_{l_3} H_{l_2} H_{l_1} + H_{l_3} H_{l_1} H_{l_2} = 0.$$

Then only the terms $H_{l_1} H_{l_2} H_{l_3}$ with $(l_1, l_2), (l_1, l_3), (l_2, l_3) \in Comm^*$ (case 1) are remained.

The upper bound of $(K+1)$-th order terms is straightforward according to the definition of E_ϵ, which are the terms not included in U_j and $-iH_l$. For the $K+2$-th order terms, $\beta_l H_l$ are already included into $\gamma_l(-iH_l)$ and the remaining terms are

$$E_{K+2} = \frac{t^{K+2}}{(K+2)!}(-iH)^{K-1} \sum_{(l_1, l_2), (l_1, l_3), (l_2, l_3) \in Comm^*} \alpha_{l_1} \alpha_{l_2} \alpha_{l_3} (-i)^3 H_{l_1} H_{l_2} H_{l_3}. \tag{65}$$

We denote the coefficient

$$\alpha_r^{(3)} = \sum_{(l_1, l_2), (l_1, l_3), (l_2, l_3) \in Comm^*} \alpha_{l_1} \alpha_{l_2} \alpha_{l_3} i H_{l_1} H_{l_2} H_{l_3}, \tag{66}$$

and we have

$$\|E_{K+2}\| \leq \frac{t^{K+2}}{(K+2)!} \|H^{K-1}\| \alpha_r^{(3)}. \tag{67}$$

When $K-1$ is even, we use the bound

$$\|H^{K-1}\| \leq \alpha_{comm}^{\frac{K-1}{2}} \tag{68}$$

in Eq. (21). We could also use the similar method to carefully calculate the rest high order part $\|E_\infty\|$, but it would not influence the error a lot. Thus we directly use the bounds in Eq. (21) and obtain

$$\|E_\infty\| \leq \frac{t^{K+3} \sqrt{\alpha_{comm}}^{K+3}}{(K+3)!} \left[\frac{\alpha}{\sqrt{\alpha_{comm}}} \left(\sum_{l \ odd, \ l \geq 0} \frac{t^l}{l!} \sqrt{\alpha_{comm}}^l \right) + \sum_{l \ even, \ l \geq 0} \frac{t^l}{l!} \sqrt{\alpha_{comm}}^l \right]. \tag{69}$$

In the modified scheme, all the low order $K-1$ order terms can be implemented as usual. To implement modified $\widetilde{R}_K$, we need to replace the K-th controlled select(H) with one controlled select(H'), where select(H') has $E+1+L$ unitaries to select, which is more complicate than select(H),

$$\text{select}(H) = \sum_{l=1}^L |l\rangle \langle l| \otimes (-iH_l),$$

$$\text{select}(H') = \sum_{l=1}^L |l\rangle \langle l| \otimes (-iH_l) + |L+1\rangle \langle L+1| \otimes (-I) \tag{70}$$

$$+ \sum_{j=1}^E |L+1+j\rangle \langle L+1+j| \otimes U_j.$$

In Ref. [17], a select(H) with L different unitaries to select requires $7.5 \cdot 2^w + 6w - 26$ CNOT gates and $7.5 \cdot 2^w + 6w - 28$ T gates, where $w = \lceil \log_2 L \rceil$. Then we can see that when $E \leq 2^w - L - 1$, select(H) and select(H') have the same gate cost. Thus, in the numerical simulation, we implement the modified schemes with $E = 0, 2^w - L - 1$, which indicates almost no additional gate cost.

D   Modified LCU method for the pair-wisely anti-commutative Hamiltonians

D.1   Proof of Lemma 1

We can prove it iteratively. The conclusion is true for $m = 1$. Now, suppose it is true for $m > 1$, we have

$$
\begin{aligned}
H^{m+1} &= H^m H \\
&= \left(\gamma_0^{(m)} I + \sum_{l=1}^{L} \gamma_l^{(m)} H_l \right) \left(\sum_{l=1}^{L} \alpha_l H_l \right) \\
&= \sum_{l=1}^{L} \gamma_l^{(m)} \alpha_l I + \gamma_0^{(m)} \sum_{l=1}^{L} \alpha_l H_l + \sum_{l_1 \neq l_2} \gamma_{l_1}^{(m)} \alpha_{l_2} H_{l_1} H_{l_2}
\end{aligned}
\tag{71}
$$

The third term is 0 due to anti-commutative relation and we have

$$
\gamma_0^{(m+1)} = \sum_l \gamma_l^{(m)} \alpha_l, \; \gamma_l^{(m+1)} = \gamma_0^{(m)} \alpha_l.
\tag{72}
$$

Thus $H^{m+1} = \gamma_0^{(m+1)} I + \sum_{l=1}^{L} \gamma_l^{(m)} H_l$.

D.2   Implementation of LCU

From Lemma 1, we can expand the evolution $U(t) = e^{-iHt}$ as

$$
U(t) = \widetilde{U}(t) = \tilde{\alpha}_0 I + \sum_l \tilde{\alpha}_l (-iH_l),
\tag{73}
$$

where

$$
\begin{aligned}
\tilde{\alpha}_0 &= 1 - \beta_s^2 t^2 / 2! + \beta_s^4 t^4 / 4! + \cdots = \cos(t\beta_s), \\
\tilde{\alpha}_l &= \alpha_l t - \alpha_l \beta_s^2 t^3 / 3! + \alpha_l \beta_s^4 t^5 / 5! + \ldots \\
&= \frac{\alpha_l}{\beta_s} \sin(t\beta_s), \; l = 1, \ldots, L. \\
\beta_s &= \sqrt{\sum_{l=1}^{L} \alpha_l^2}.
\end{aligned}
\tag{74}
$$

Now we show how to implement the evolution via the LCU method. We show two different methods here. The first method is via direct implementation of $U(t)$ via LCU. Similar to the original LCU method, we need oracles G, select(U),

$$
\begin{aligned}
G \left| 0 \right\rangle &= \frac{1}{\sqrt{s}} \sum_j \sqrt{|\tilde{\alpha}_l|} \left| j \right\rangle, \\
\text{select}(U) &= \sum_l \left| l \right\rangle \left\langle l \right| \otimes U_l, \\
W &= (G^\dagger \otimes I) \text{select}(U)(G \otimes I), \\
s &= \sum_l |\tilde{\alpha}_l| = |\cos(t\beta_s)| + \frac{\alpha}{\beta_s} |\sin(t\beta_s)|.
\end{aligned}
\tag{75}
$$

Here we choose $U_0 = I$ or $-I$, $U_l = -iH_l$ or iH_l according to the sign of $\tilde{\alpha}_l$. According to Ref. [8], select(U) can be implemented via select(U_l) one by one (totally $L + 1$). And each select(U_l) requires

$O(n+\log L)$ gates where n is the number of qubits in the target system. Thus we need $O(L(n+\log L))$ gates to implement W. Similar to the original method, we also need multiple use of W and reflection to boost the coefficient before $|0\rangle \otimes U(t)|\Psi\rangle$. According to Ref. [7], we require $O(s)$ number of repetition use of W. Due to $1 \le s \le 1+\sqrt{L}$, the total gate complexity is $O(L^{\frac{3}{2}}(n+\log L))$, which is only related to the number of terms used in LCU L, not related to t and ε.

The second method uses a few different segments to simulate the whole evolution. From Eq. (38), when $s = 1$ we do not need to use the oblivious amplitude amplification, $U(t)$ can be implemented by

$$W(|0\rangle \otimes |\Psi\rangle) = |0\rangle \otimes U(t)|\Psi\rangle. \tag{76}$$

Thus we choose the first segment as $t_1 = \lfloor \frac{t\beta_s}{\pi} \rfloor \frac{\pi}{\beta_s}$ such that $s = 1$. The rest evolution time is $t_2 = t - t_1 \le \frac{\pi}{\beta_s}$ and the corresponding s can still be very large. When $\frac{\alpha^2}{\beta_s^2} \ge 3$, we choose another simulation time t_{seg} as

$$t_{seg} = \frac{\arcsin\left[\frac{2\frac{\alpha}{\beta_s} - \sqrt{\frac{\alpha^2}{\beta_s^2} - 3}}{1 + \frac{\alpha^2}{\beta_s^2}}\right]}{\beta_s} \tag{77}$$

such that

$$s = |\cos(t_{seg}\beta_s)| + \frac{\alpha}{\beta_s}|\sin(t_{seg}\beta_s)| = 2. \tag{78}$$

And t_2 is divided into r segments $r = \lfloor \frac{t_2}{t_{seg}} \rfloor$, $t_2 = rt_{seg} + t_{rest}, t_{rest} \le t_{seg}$. In the rest time, we have

$$1 < s = |\cos(t_{rest}\beta_s)| + \frac{\alpha}{\beta_s}|\sin(t_{rest}\beta_s)| < 2. \tag{79}$$

Thus we need $r + 2$ segments, where r can be bound as

$$r \le \frac{\pi}{\arcsin\left[\frac{2\frac{\alpha}{\beta_s} - \sqrt{\frac{\alpha^2}{\beta_s^2} - 3}}{1 + \frac{\alpha^2}{\beta_s^2}}\right]} = \mathcal{O}(\sqrt{L}) \tag{80}$$

The last equation is due to $\frac{\alpha^2}{\beta_s^2} \le L$.

When $\frac{\alpha^2}{\beta_s^2} < 3$, we also have $1 < s < 2$. For all the segments where $s < 2$, we can use another aniclla to convert $s < 2$ into the $s = 2$ case [8, 17]. And all the $s = 2$ segments can be implemented via the oblivious amplitude amplification as the usual LCU method. Consequently, the upper bound of gate cost is $r\mathcal{O}(L(n+\log L)) = \mathcal{O}(L^{\frac{3}{2}}(n+\log L))$. We could implement this digital simulation efficiently without algorithmic error via standard LCU method.

D.3 Proof of Corollary 2

Here we consider the case where the Hamiltonian H is close to the perfectly anti-commutative,

$$H^2 = \beta_s^2 I + V, \quad \|V\| \le \varepsilon_A. \tag{81}$$

Then the unitary evolution can be expressed as

$$\begin{aligned}
U_0(t) &= \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!} \\
&= \sum_{even\ k \ge 0} \frac{t^k}{k!}(-\beta_s^2 I - V)^{\frac{k}{2}} + \sum_{odd\ k \ge 1} \frac{t^k}{k!}(-\beta_s^2 I - V)^{\lfloor \frac{k}{2} \rfloor} \sum_l (-iH_l).
\end{aligned} \tag{82}$$

Note that here V is not unitary but the sum of a few different unitaries. Here we also implement the operator $\widetilde{U}(t)$ in the perfect case in Eq. (36)

$$\widetilde{U}(t) = \tilde{\alpha}_0 I + \sum_l \tilde{\alpha}_l (-iH_l),$$

(83)

and the error can be bound by

$$
\begin{aligned}
&\left\| \widetilde{U}(t) - \sum_{k=0}^{\infty} \frac{(-itH)^k}{k!} \right\| \\
&\leq \sum_{even\ k \geq 2} \frac{t^k}{k!} [(\beta_s^2 + \varepsilon_A)^{\frac{k}{2}} - \beta_s^k] + \sum_{odd\ k \geq 3} \frac{t^k}{k!} \alpha [(\beta_s^2 + \varepsilon_A)^{\lfloor \frac{k}{2} \rfloor} - \beta_s^{k-1}] \\
&\leq \frac{1}{2} \left[\left(e^{t\sqrt{\beta_s^2 + \varepsilon_A}} + e^{-t\sqrt{\beta_s^2 + \varepsilon_A}} \right) - \left(e^{t\beta_s} + e^{-t\beta_s} \right) \right] \\
&\quad + \frac{1}{2} \left[\frac{\alpha}{\sqrt{\beta_s^2 + \varepsilon_A}} \left(e^{t\sqrt{\beta_s^2 + \varepsilon_A}} - e^{-t\sqrt{\beta_s^2 + \varepsilon_A}} \right) - \frac{\alpha}{\beta_s} \left(e^{t\beta_s} - e^{-t\beta_s} \right) \right],
\end{aligned}
$$

(84)

which is roughly $\mathcal{O}(t^2 \varepsilon_A) + \mathcal{O}(t^3 \varepsilon_A \alpha)$ with a small t. The first inequality is because high order of $\beta_s^2 I$ are all included in the coefficient γ_0.

References

[1] Alán Aspuru-Guzik, Anthony D Dutoi, Peter J Love, and Martin Head-Gordon. Simulated quantum computation of molecular energies. *Science*, 309(5741):1704–1707, 2005. DOI: https://doi.org/10.1126/science.1113479. URL https://science.sciencemag.org/content/309/5741/1704.

[2] Ryan Babbush, Peter J Love, and Alán Aspuru-Guzik. Adiabatic quantum simulation of quantum chemistry. *Scientific reports*, 4:6603, 2014. DOI: https://doi.org/10.1038/srep06603. URL https://rdcu.be/cv1RR.

[3] Ryan Babbush, Jarrod McClean, Dave Wecker, Alán Aspuru-Guzik, and Nathan Wiebe. Chemical basis of trotter-suzuki errors in quantum chemistry simulation. *Phys. Rev. A*, 91:022311, Feb 2015. DOI: https://doi.org/10.1103/PhysRevA.91.022311. URL https://link.aps.org/doi/10.1103/PhysRevA.91.022311.

[4] R Barends, L Lamata, J Kelly, L García-Álvarez, AG Fowler, A Megrant, E Jeffrey, TC White, D Sank, JY Mutus, et al. Digital quantum simulation of fermionic models with a superconducting circuit. *Nature communications*, 6:7654, 2015. DOI: https://doi.org/10.1038/ncomms8654. URL https://rdcu.be/cv1R9.

[5] Dominic W Berry and Andrew M Childs. Black-box hamiltonian simulation and unitary implementation. *Quantum Information & Computation*, 12(1-2):29–62, 2012. URL https://dl.acm.org/doi/10.5555/2231036.2231040.

[6] Dominic W Berry, Graeme Ahokas, Richard Cleve, and Barry C Sanders. Efficient quantum algorithms for simulating sparse hamiltonians. *Communications in Mathematical Physics*, 270(2): 359–371, 2007. DOI: https://doi.org/10.1007/s00220-006-0150-x.

[7] Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. Exponential improvement in precision for simulating sparse hamiltonians. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 283–292, 2014. DOI: https://doi.org/10.1145/2591796.2591854.

[8] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Simulating hamiltonian dynamics with a truncated taylor series. *Phys. Rev. Lett.*, 114:090502, Mar 2015. DOI: https://doi.org/10.1103/PhysRevLett.114.090502. URL https://link.aps.org/doi/10.1103/PhysRevLett.114.090502.

Accepted in ⟨Quantum 2021-08-24, click title to verify. Published under CC-BY 4.0.

17

[9] Sergey Bravyi. Monte carlo simulation of stoquastic hamiltonians. *Quantum Information & Computation*, 15(13-14):1122–1140, 2015. URL `https://dl.acm.org/doi/10.5555/2871363.2871366`.

[10] Sergey Bravyi and David Gosset. Polynomial-time classical simulation of quantum ferromagnets. *Phys. Rev. Lett.*, 119:100503, Sep 2017. DOI: https://doi.org/10.1103/PhysRevLett.119.100503. URL `https://link.aps.org/doi/10.1103/PhysRevLett.119.100503`.

[11] Kenneth R. Brown, Robert J. Clark, and Isaac L. Chuang. Limitations of quantum simulation examined by simulating a pairing hamiltonian using nuclear magnetic resonance. *Phys. Rev. Lett.*, 97:050504, Aug 2006. DOI: https://doi.org/10.1103/PhysRevLett.97.050504. URL `https://link.aps.org/doi/10.1103/PhysRevLett.97.050504`.

[12] Earl Campbell. Shorter gate sequences for quantum computing by mixing unitaries. *Phys. Rev. A*, 95:042306, Apr 2017. DOI: https://doi.org/10.1103/PhysRevA.95.042306. URL `https://link.aps.org/doi/10.1103/PhysRevA.95.042306`.

[13] Earl Campbell. Random compiler for fast hamiltonian simulation. *Phys. Rev. Lett.*, 123:070503, Aug 2019. DOI: https://doi.org/10.1103/PhysRevLett.123.070503. URL `https://link.aps.org/doi/10.1103/PhysRevLett.123.070503`.

[14] Chi-Fang Chen, Richard Kueng, Joel A Tropp, et al. Concentration for random product formulas. *arXiv preprint arXiv:2008.11751*, 2020.

[15] Andrew M. Childs and Yuan Su. Nearly optimal lattice simulation by product formulas. *Phys. Rev. Lett.*, 123:050503, Aug 2019. DOI: https://doi.org/10.1103/PhysRevLett.123.050503. URL `https://link.aps.org/doi/10.1103/PhysRevLett.123.050503`.

[16] Andrew M Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Information & Computation*, 12(11-12):901–924, 2012. URL `https://dl.acm.org/doi/10.5555/2481569.2481570`.

[17] Andrew M. Childs, Dmitri Maslov, Yunseong Nam, Neil J. Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, 2018. ISSN 0027-8424. DOI: https://doi.org/10.1073/pnas.1801723115. URL `https://www.pnas.org/content/115/38/9456`.

[18] Andrew M. Childs, Aaron Ostrander, and Yuan Su. Faster quantum simulation by randomization. *Quantum*, 3:182, September 2019. ISSN 2521-327X. DOI: https://doi.org/10.22331/q-2019-09-02-182. URL `https://doi.org/10.22331/q-2019-09-02-182`.

[19] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu. Theory of trotter error with commutator scaling. *Phys. Rev. X*, 11:011020, Feb 2021. DOI: https://doi.org/10.1103/PhysRevX.11.011020. URL `https://link.aps.org/doi/10.1103/PhysRevX.11.011020`.

[20] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6):467–488, Jun 1982. ISSN 1572-9575. DOI: https://doi.org/10.1007/BF02650179. URL `https://doi.org/10.1007/BF02650179`.

[21] Markus Heyl, Philipp Hauke, and Peter Zoller. Quantum localization bounds trotter errors in digital quantum simulation. *Science advances*, 5(4):eaau8342, 2019. DOI: https://doi.org/10.1126/sciadv.aau8342. URL `https://advances.sciencemag.org/content/5/4/eaau8342`.

[22] P. Jordan and E. Wigner. Über das Paulische Äquivalenzverbot. *Zeitschrift für Physik*, 47(9):631–651, September 1928. ISSN 0044-3328. DOI: https://doi.org/10.1007/BF01331938. URL `https://doi.org/10.1007/BF01331938`.

[23] Stephen P Jordan, Keith SM Lee, and John Preskill. Quantum algorithms for quantum field theories. *Science*, 336(6085):1130–1133, 2012. DOI: https://doi.org/10.1126/science.1217069. URL `https://science.sciencemag.org/content/336/6085/1130`.

[24] Seth Lloyd. Universal quantum simulators. *Science*, 273:1073–1078, Aug 1996. DOI: https://doi.org/10.1126/science.273.5278.1073. URL `http://science.sciencemag.org/content/273/5278/1073/tab-article-info`.

[25] Guang Hao Low and Isaac L. Chuang. Optimal hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118:010501, Jan 2017. DOI: https://doi.org/10.1103/PhysRevLett.118.010501. URL https://link.aps.org/doi/10.1103/PhysRevLett.118.010501.

[26] Guang Hao Low and Isaac L. Chuang. Hamiltonian Simulation by Qubitization. *Quantum*, 3: 163, July 2019. ISSN 2521-327X. DOI: https://doi.org/10.22331/q-2019-07-12-163. URL https://doi.org/10.22331/q-2019-07-12-163.

[27] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C. Benjamin, and Xiao Yuan. Quantum computational chemistry. *Rev. Mod. Phys.*, 92:015003, Mar 2020. DOI: https://doi.org/10.1103/RevModPhys.92.015003. URL https://link.aps.org/doi/10.1103/RevModPhys.92.015003.

[28] Jarrod R McClean, Nicholas C Rubin, Kevin J Sung, Ian D Kivlichan, Xavier Bonet-Monroig, Yudong Cao, Chengyu Dai, E Schuyler Fried, Craig Gidney, Brendan Gimby, Pranav Gokhale, Thomas Häner, Tarini Hardikar, Vojtěch Havlíček, Oscar Higgott, Cupjin Huang, Josh Izaac, Zhang Jiang, Xinle Liu, Sam McArdle, Matthew Neeley, Thomas O'Brien, Bryan O'Gorman, Isil Ozfidan, Maxwell D Radin, Jhonathan Romero, Nicolas P D Sawaya, Bruno Senjean, Kanav Setia, Sukin Sim, Damian S Steiger, Mark Steudtner, Qiming Sun, Wei Sun, Daochen Wang, Fang Zhang, and Ryan Babbush. OpenFermion: the electronic structure package for quantum computers. *Quantum Science and Technology*, 5(3):034014, jun 2020. DOI: https://doi.org/10.1088/2058-9565/ab8ebc. URL https://doi.org/10.1088/2058-9565/ab8ebc.

[29] Richard Meister, Simon C Benjamin, and Earl T Campbell. Tailoring term truncations for electronic structure calculations using a linear combination of unitaries. *arXiv preprint arXiv:2007.11624*, 2020.

[30] R. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme. Simulating physical phenomena by quantum networks. *Phys. Rev. A*, 65:042323, Apr 2002. DOI: https://doi.org/10.1103/PhysRevA.65.042323. URL https://link.aps.org/doi/10.1103/PhysRevA.65.042323.

[31] Yuan Su, Hsin-Yuan Huang, and Earl T. Campbell. Nearly tight Trotterization of interacting electrons. *Quantum*, 5:495, July 2021. ISSN 2521-327X. DOI: https://doi.org/10.22331/q-2021-07-05-495. URL https://doi.org/10.22331/q-2021-07-05-495.

[32] Masuo Suzuki. Generalized trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems. *Comm. Math. Phys.*, 51: 183–190, 1976. DOI: https://doi.org/10.1007/BF01609348. URL https://rdcu.be/cv10X.

[33] Masuo Suzuki. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Journal of Mathematical Physics*, 32(2):400–407, 1991. DOI: https://doi.org/10.1063/1.529425.

[34] Minh C. Tran, Su-Kuan Chu, Yuan Su, Andrew M. Childs, and Alexey V. Gorshkov. Destructive error interference in product-formula lattice simulation. *Physical Review Letters*, 124(22), Jun 2020. ISSN 1079-7114. DOI: https://doi.org/10.1103/physrevlett.124.220502. URL http://dx.doi.org/10.1103/PhysRevLett.124.220502.

[35] Dave Wecker, Bela Bauer, Bryan K. Clark, Matthew B. Hastings, and Matthias Troyer. Gate-count estimates for performing quantum chemistry on small quantum computers. *Phys. Rev. A*, 90:022305, Aug 2014. DOI: https://doi.org/10.1103/PhysRevA.90.022305. URL https://link.aps.org/doi/10.1103/PhysRevA.90.022305.

[36] Dave Wecker, Matthew B. Hastings, Nathan Wiebe, Bryan K. Clark, Chetan Nayak, and Matthias Troyer. Solving strongly correlated electron models on a quantum computer. *Phys. Rev. A*, 92: 062318, Dec 2015. DOI: https://doi.org/10.1103/PhysRevA.92.062318. URL https://link.aps.org/doi/10.1103/PhysRevA.92.062318.