Evaluation of gate numbers for ground state energy calculations using higher-order product formulae

Hiromu Abe,[1] Kosuke Mitarai,[1, 2] Keita Kanno,[3] and Ryosuke Kimura[1]

[1] *Graduate School of Engineering Science, The University of Osaka, 1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan*
[2] *Center for Quantum Information and Quantum Biology, The University of Osaka, 1-2 Machikaneyama, Toyonaka, Osaka 560-0043, Japan*
[3] *QunaSys Inc., Aqua Hakusan Building 9F, 1-13-7 Hakusan, Bunkyo, Tokyo 113-0001, Japan*

Trotter expansion and quantum phase estimation (QPE) are powerful quantum algorithms for computing ground state energies of molecular Hamiltonians in quantum chemistry. In this paper, we estimate the overall computational cost of the combined Trotter–QPE approach for hydrogen chains with a prescribed target energy error, using both the total number of Pauli rotations (total gate count) and the depth of R_Z-rotation layers as cost metrics. For evaluations up to H_{15}, when the required error is one order of magnitude tighter than chemical accuracy, namely 0.16 mHa, the eighth-order product formula by Morales *et al.* minimizes the total gate count. In contrast, for the R_Z-layer depth, the preferred product formula depends on the qubit count: Morales *et al.*'s eighth-order formula gives the smallest depth up to around 50 qubits, whereas the conventional fourth-order formula tends to yield a smaller depth for larger qubit counts. We also observe that, near chemical accuracy, the tenth-order formula can require a larger total gate count than the eighth-order formula. Furthermore, we propose a new fourth-order product formula based on the methodology of Morales *et al.*; at chemical accuracy (about 1.6 mHa), it yields a smaller total gate count than Morales *et al.*'s eighth-order formula for all hydrogen chains up to H_{15}, and it also reduces the depth of R_Z-rotation layers.

I. INTRODUCTION

Quantum simulation of quantum mechanical systems is widely recognized as one of the most promising applications of quantum computers. It has the potential to solve problems that are intractable on classical computers, ranging from quantum chemistry to condensed matter and high-energy physics. Among the various challenges in quantum simulation, obtaining the eigenvalues of the system Hamiltonian is particularly important, as they correspond to the fundamental energy levels of the system. Classical algorithms typically require computational resources that grow exponentially with system size. In contrast, quantum algorithms can find these eigenvalues in polynomial time [1, 2].

To this end, a variety of quantum algorithms have been developed. Product formulae, also known as Trotter–Suzuki decompositions, have served as fundamental methods for simulating Hamiltonian time evolution [3, 4]. More recently, advanced methods based on block encoding and quantum singular value transformation (QSVT) have been introduced [5, 6]. These modern approaches exponentially improve the scaling of approximation error with respect to evolution time and are therefore considered highly promising for high-precision evaluation. However, the practical advantage of such sophisticated algorithms has strongly relied on the assumption that non-Clifford operations, such as T gates, are extremely costly due to the substantial resource overhead of magic-state distillation [7].

This paradigm is beginning to shift with recent advances in fault-tolerant quantum computing architectures. Proposals such as magic-state cultivation [8], zero-level distillation [9], and the STAR architecture [10] suggest the possibility of generating magic states locally with significantly fewer resources than before. In such hardware-efficient regimes, algorithms that can effectively exploit the locality of the Hamiltonian can become more resource-efficient than QSVT, which requires more global structures. These changes strongly motivate a re-evaluation of the performance of product formulae for practical quantum simulation.

In this study, we focus on the family of higher-order product formulae proposed by Morales *et al.* in 2022 [11]. Their work builds on the classical constructions by Yoshida in the 1990s [12], while numerically optimizing the formula parameters with modern computational resources to minimize leading error terms. The central aim of this work is to quantitatively assess the practicality of applying these formulae to the ground state energy calculation of molecular Hamiltonians.

We analyze errors induced by product formulae numerically and evaluate the resources required to obtain ground state energies of hydrogen chains. A perturbative strategy to estimate the eigenvalue errors between the true time evolution and its product-formula approximation, which can be of independent interest, allows us to evaluate these errors up to the 30-qubit scale. We then extrapolate the results to estimate the gate cost for larger systems. Our analysis yields several findings. When aiming at a target precision of 0.16 mHa, which is one order of magnitude tighter than chemical accuracy, the eighth-order product formula by Morales *et al.* minimizes the total number of Pauli rotations for all qubit counts up to 100, as shown in Fig. 1. In contrast, when using the depth of R_Z-rotation layers as the cost metric, Fig. 2 shows that Morales *et al.*'s eighth-order formula gives the smallest depth up to 50 qubits, while the conventional

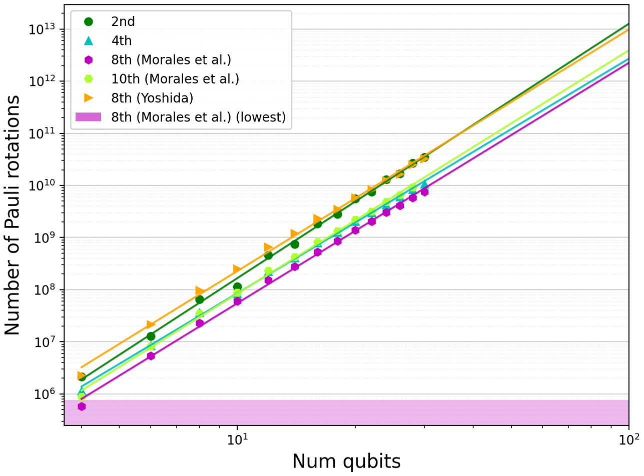


FIG. 1. For H-chains at the target energy error $\varepsilon_{\mathrm{E}} = 1.6 \times 10^{-4}$ Ha, we plot the total gate count F (number of Pauli rotations) as dots for each qubit count. The colored straight lines show the fitted scaling on logarithmic scales for both axes. Legend: second order (green), fourth order (cyan), eighth order by Morales *et al.* (magenta), tenth order by Morales *et al.* (greenyellow). The color band below the plot indicates, for each qubit count, which product formula attains the smallest F.

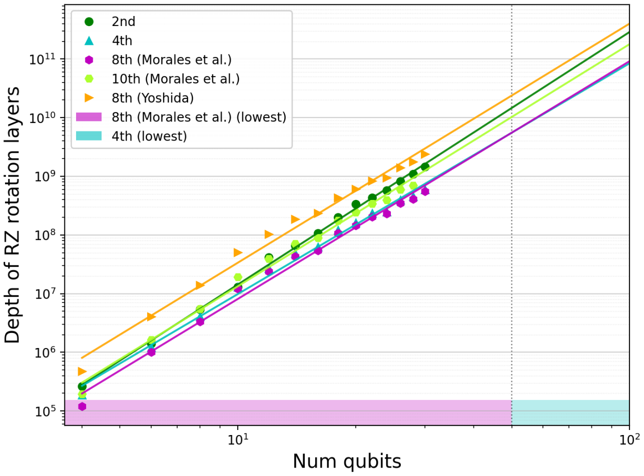


FIG. 2. For H-chains at the target energy error $\varepsilon_{\mathrm{E}} = 1.6 \times 10^{-4}$ Ha, we plot the depth of R_Z-rotation layers as dots for each qubit count. The colored straight lines show the fitted scaling on logarithmic scales for both axes. Legend: second order (green), fourth order (cyan), eighth order by Morales *et al.* (magenta), tenth order by Morales *et al.* (greenyellow), eighth order by Yoshida (orange). The color band below the plot indicates, for each qubit count, which product formula attains the smallest depth of R_Z-rotation layers.

fourth-order product formula becomes the lowest-depth choice for larger qubit counts. Moreover, motivated by an intuition obtained during the analysis, we construct a new fourth-order formula and find that, at the same target precision of 0.16 mHa, it yields a smaller cost in terms of both the total number of rotations and the R_Z-layer depth than the two baseline formulae (Morales *et al.*'s eighth-order formula and the conventional fourth-order formula), as shown in Figs. 3 and 4.

II. THEORETICAL BACKGROUND

First, in Sec. II.1, we review previous studies relevant to this work. Then, in Sec. II.2, we present our new considerations beyond these prior studies.

II.1. Quantum phase estimation and product formulae

II.1.1. Quantum phase estimation

Quantum phase estimation (QPE) [1] is an algorithm to estimate eigenphases φ of a unitary U using controlled-version of U. Let $\hat{\varphi}$ denote the phase estimated by QPE and φ the true phase to be estimated. We define the estimation error of QPE as

$$\varepsilon = \left|\hat{\varphi} - \varphi\right| .$$

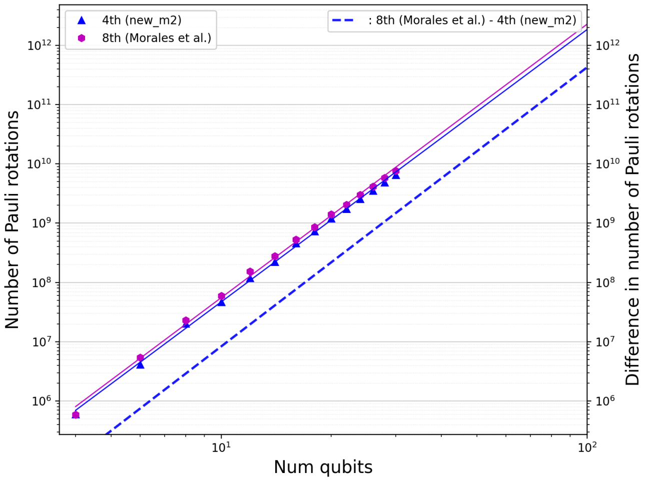


FIG. 3. As in Fig. 1, we plot the total gate count F at the target energy error $\varepsilon_{\mathrm{E}} = 1.6 \times 10^{-4}$ Ha together with the fitted scaling on log–log axes. We compare the eighth-order product formula by Morales *et al.* (magenta, “8th (Morales *et al.*)”) with our newly constructed fourth-order product formula (blue, “4th (new_m2)”). The dashed curve shows the absolute difference between the two estimated depths, plotted on the right axis.

The number of unitary applications required to estimate the phase with error ε, denoted $M(\varepsilon)$ is generally given by

$$M(\varepsilon) = \frac{\beta}{\varepsilon}, \tag{1}$$

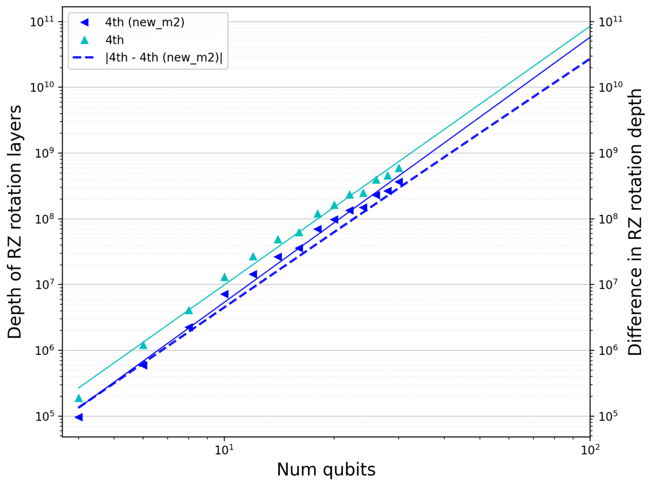


FIG. 4. As in Fig. 1, we plot the depth of R_Z-rotation layers at the target energy error $\varepsilon_{\mathrm{E}} = 1.6 \times 10^{-4}$ Ha together with the fitted scaling on log–log axes. We compare our new fourth-order product formula (blue, "4th (new_m2)") with the original fourth-order formula (cyan, "4th"). The dashed curve shows the absolute difference between the two estimated depths, plotted on the right axis.

for some constant β. A theoretical upper-bound on β is $\beta \leq 16\pi$ [13]. In practice, however, a much smaller constant factor suffices to achieve error ε. For resource analysis performed later, we heuristically find $\beta = 1.20$ from numerical evaluation presented in Appendix A.

II.1.2. Product formulae

Suppose a Hamiltonian H is expressed as a linear combination of J terms H_j with $j \in \{1, \ldots, J\}$:

$$H = \sum_{j=1}^{J} H_j \,.$$

Product formulae approximate the time-evolution operator e^{iHt} via the product of $e^{iH_j t}$. The second-order one is probably the most popular and approximates the time evolution as

$$\begin{aligned} S_2(t) &:= e^{iHt} \\ &= \prod_{p=1}^{J-1} e^{\frac{1}{2} i H_p t} e^{i H_J t} \prod_{q=1}^{J-1} e^{\frac{1}{2} i H_{J-q} t} + \mathcal{O}(t^3)\,. \end{aligned} \tag{2}$$

A common method to construct higher-order product formulae is Suzuki's fractal decomposition [14]. First, let $S_2(t)$ denote the second-order product formula in Eq. (2). A product formula of order $k = 2\kappa$ with $\kappa \in \mathbb{N}$ can be constructed recursively as

$$S_{2\kappa}(t) = S_{2\kappa-2}(s_\kappa t) S_{2\kappa-2}((1-2s_\kappa)t) S_{2\kappa-2}(s_\kappa t)\,, \tag{3}$$

where $s_\kappa = 1/(2 - 2^{1/(2\kappa-1)})$. The total number of matrix exponentials N_{exp} in $S_{2\kappa}$ is

$$N_{\mathrm{exp}} = 2(J-1)3^{\kappa-1} + 1\,. \tag{4}$$

Thus, the number of matrix exponentials grows exponentially with the order of the product formula.

To address this exponential growth, Yoshida proposed an alternative construction [12]:

$$S^{(m)}(t) = \left(\prod_{j=1}^{m} S_2(w_{m-j+1} t) \right) S_2(w_0 t) \left(\prod_{j=1}^{m} S_2(w_j t) \right)\,, \tag{5}$$

where $w_j \in \mathbb{R}$ for $j = 0, 1, \ldots, m$ are parameters. The total number of matrix exponentials N_{exp} in a formula constructed this way is

$$N_{\mathrm{exp}} = (4m+2)(J-1) + 1. \tag{6}$$

The coefficients w_j are determined by recursively expanding Eq. (5) using the Baker–Campbell–Hausdorff (BCH) formula [15] and imposing conditions such that all terms below a certain order k vanish. Yoshida constructed product formulae up to the eighth order [12]. In this eighth-order case, $m = 7$.

Building on Yoshida's approach, Morales *et al.* employed numerical optimization to identify new coefficients and constructed both eighth- and tenth-order formulae [11]. Their eighth-order product formula achieves smaller errors than Yoshida's, and they also introduced a new tenth-order formula.

II.2. Analysis of the total computational cost

In this work, we estimate the computational cost required to obtain the ground state energy with target error ε_{E} using QPE and product formulae. To this end, here we provide closed-form expression to calculate the number of exponentials needed to achieve the precision.

Let E' be the eigenenergy of the approximated time-evolution operator constructed by a pth-order product formula consisting of N_{exp} matrix exponentials and E be its corresponding true value of H. We define α and p as

$$E = E' + \alpha t^p, \quad \alpha, p \in \mathbb{R}\,. \tag{7}$$

The total number of exponentials F required to achieve the target error ε_{E} is given by

$$F := \beta N_{\mathrm{exp}} \left(\frac{1+p}{p\varepsilon_E} \right) \left(\frac{\alpha(1+p)}{\varepsilon_E} \right)^{\frac{1}{p}} \tag{8}$$

The derivation of Eq. (8) is provided in Appendix E.

We next analyze the partial derivatives of F with respect to α and p. For α, p, ε_E, J in the ranges $\alpha \in [10^{-16}, 10^{-3}]$, $p \in [2, 10]$, $\varepsilon_E \in [10^{-5}, 10^{-3}]$, and $J \gg 1$, we find $\frac{\partial F}{\partial \alpha} > 0$. Thus, smaller α reduces the gate count in practice. Moreover, when $\frac{e^{8/5}}{3} > \frac{\alpha}{\varepsilon_E} > 0$, we also obtain $\frac{\partial F}{\partial p} > 0$. Therefore, increasing p suppresses product-formula error but increases the gate count, leading to a trade-off. While increasing p decreases the number of repetitions of the unitary required by QPE, it simultaneously increases the number of gates needed to construct each unitary. As a result, the contribution from the increased gate count per unitary outweighs the reduction from fewer repetitions. Hence, both reducing α and increasing p suppress errors, but reducing α decreases the total gate count, whereas increasing p increases it—an apparently counterintuitive outcome.

The condition $\frac{\partial F}{\partial \alpha} > 0$ always holds, whereas $\frac{\partial F}{\partial p} > 0$ holds only if $\frac{e^{8/5}}{3} > \frac{\alpha}{\varepsilon_E} > 0$. Consequently, when ε_E and α satisfy this condition, one can reduce the total gate count more effectively by decreasing α rather than increasing p. Appendix B presents the derivations of $\frac{\partial F}{\partial \alpha} > 0$ and $\frac{\partial F}{\partial p} > 0$.

III. EVALUATION METHOD

In this study, we compare and evaluate the total computational cost of various product formulae when calculating the ground state energy of molecular Hamiltonians using phase estimation with a specified target error. In other words, we compare F defined in Eq. (8) across different product formulae. In addition, to capture a compilation-aware cost relevant to fault-tolerant implementations such as the STAR architecture, we also estimate the depth of R_Z-rotation layers, defined as the number of sequential R_Z layers after compiling the grouped evolutions into a Z-diagonal form.

The formulae considered are the second-order, fourth-order [16], eighth-order, and tenth-order product formulae. Among the eighth-order cases, we include both Yoshida's construction and that of Morales *et al.*, and we also include the tenth-order formula constructed by Morales *et al.* For the second-order case we use Eq. (2), and for the higher-order formulae we adopt the coefficients listed in Tables III and IV in Appendix D.

We choose one-dimensional hydrogen chains $\mathrm{H_n}$ with $n = 2, \ldots, 15$ as benchmark systems. The internuclear distances are fixed at 1.0 Å, and we use the STO-3G basis set. For H_{15}, we only compare the second-, fourth-, and eighth-order formulae, due to its high computational demand.

The overall evaluation procedure is as follows:

1. Perform restricted Hartree-Fock (HF) calculations using `PySCF` [17, 18], and construct the second-quantized Hamiltonian H using the HF basis. We then translate it into Pauli operators using Jordan-Wigner transformation [19] with `OpenFermion` [20, 21]. We also calculate the true ground state energy E by performing full configuration interaction calculation using `PySCF`.
2. Evaluate the error δE between E and E' for multiple values of t, and fit the error to αt^p to determine α and p. This is done by the perturbative method described in Appendix F.
3. Calculate F via Eq. (8) and the depth of R_Z-rotation layers, and compare these costs for each product formula as functions of the target error ε_E.
4. For one-dimensional hydrogen chains, we place the number of qubits on the horizontal axis and each cost metric on the vertical axis, fit the data on a log-log plot for systems that achieve the target error ε_E, evaluate the scaling of F and the depth of R_Z-rotation layers with system size, and use those scalings to extrapolate the costs to larger qubit counts.

Note that, at step (1), we group the fermionic Hamiltonian into mutually commuting terms to reduce the error of product formulae. Specifically,

$$H = \sum_i H_i, \quad H_i = \sum_j P_{ij}, \quad [P_{ij}, P_{ij'}] = 0\,,$$

where all Pauli terms within the same group H_i commute. Here, j' denotes an index different from j within the group H_i, that is, $j' \neq j$. We then treat each grouped set H_i as a primitive operator and decompose the time evolution according to e.g. (5). For grouping, we use the method based on the finite projective plane proposed by Inoue [22]. The number of groups obtained via this method is $O(n^2)$ for n-spin-orbital systems.

To estimate the depth of R_Z-rotation layers, we compile each commuting group H_i into a Z-diagonal form. Since all terms within H_i mutually commute, they admit a common Clifford basis-change circuit that simultaneously diagonalizes the entire group. For the groups obtained by the Inoue grouping [22], this Clifford circuit maps all terms in H_i to sums of at most two-body Z operators. We implement the corresponding diagonal evolution using R_Z rotations generated by these Z and ZZ terms, and define the depth of R_Z-rotation layers as the minimum number of sequential R_Z layers required when rotations acting on disjoint sets of qubits are executed in parallel. We evaluate this layer depth by constructing a conflict graph for the Z-type terms and applying a graph-coloring procedure. In this estimate, we neglect the depth of the Clifford basis-change circuit. The explicit basis-change construction and the resulting Z-diagonal forms used in our implementation are provided in Appendix C.

III.1. Extraction of α and p in the error term

When the eigenvalue error of a product formula is expressed as in Eq. (7), we determine α and p for each formula. We plot the error between E and E', defined as $\varepsilon_\mathrm{PF} = |E - E'|$, as a function of t. This plot is then fitted to the error model

$$\log \varepsilon_\mathrm{PF} = \log \alpha + p \log t\,,$$

where the intercept is included in the fitting procedure.

To calculate the total gate count with Eq. (8) using the obtained α and p, we assume that the error of the product formula can be written as αt^p for arbitrary t. Under this assumption, the optimal t that minimizes the total gate count is

$$t = \left(\frac{\varepsilon_{\mathrm{E}}}{\alpha(p+1)} \right)^{\frac{1}{p}} . \tag{9}$$

This implies that the optimal t minimizing the total gate count can be calculated from α and p obtained in any range of t, even if the t in Eq. (9) does not lie within the fitting interval. The derivation of Eq. (9) is given in Appendix E.

By following this procedure, we determine α and p for each product formula and estimate the total gate count required to achieve a target error ε_E. An illustrative example of this process for the case of H_2 is presented in Appendix G.

IV. EVALUATION RESULTS

We evaluate one-dimensional hydrogen chains H_n with $n = 2, \ldots, 15$. For H_{15}, we omit the tenth-order product formula because its decomposition count becomes prohibitively large and the evaluation cost is not practical, so we compare only the second-, fourth-, and eighth-order formulas. We set the target energy error to $\varepsilon_{\mathrm{E}} = 1.6 \times 10^{-4}\,\mathrm{Ha}$, which is one-tenth of chemical accuracy.

The results are shown in Fig. 1. Across all evaluated H-chains up to H_{15}, the eighth-order product formula by Morales *et al.* yields the smallest total gate count F. The same eighth-order formula by Morales *et al.* also minimizes F for all qubit counts up to 100 in our extrapolation.

We also evaluate a compilation-aware cost metric, the depth of R_Z-rotation layers, defined in Sec. III. The results are shown in Fig. 2. For the evaluated systems, the eighth-order product formula by Morales *et al.* attains the smallest depth. In our extrapolation, however, the fourth-order formula becomes the lowest-depth choice beyond the crossover region indicated by the shaded band in Fig. 2.

IV.1. Scaling of error coefficient α

We analyze how the error coefficient α changes with the number of spin orbitals by fitting it to three models: $\alpha = Cn^r$, $\alpha = C(\log_2 n)^r$, and $\alpha = C(\log_2(\log_2 n))^r$. The coefficients of determination R^2 obtained from these fits are summarized in Tab. I. As shown in Tab. I, for the eighth- and tenth-order product formulae constructed by Morales *et al.*, the fitting model $\alpha = C(\log_2(\log_2 n))^r$ yields larger R^2 values.

Tab. VI shows the values of α obtained by fitting $\log \varepsilon_{\mathrm{PF}} = \log \alpha + p \log t$ while fixing p to the order of the product formula used. Comparing the same-order formulae in Tab. VI, we find that for the eighth-order case, α ranges from 10^{-4} to 10^{-2} for Yoshida's formula, whereas it ranges from 10^{-12} to 10^{-8} for Morales *et al.*'s formula. This indicates that Morales *et al.*'s formula achieves significantly smaller α.

V. DISCUSSION

V.1. Relation between the error term αt^p and the cost metrics

We first discuss possible factors that reduce the total gate count when comparing product formulae. As shown in Fig. 1, even for the same eighth-order case, Morales *et al.*'s formula yields a smaller total gate count F than Yoshida's. According to Eq. (6), the decomposition count per step is $m = 7$ for Yoshida's formula and $m = 8$ for Morales *et al.*'s formula, so Yoshida's requires fewer decompositions per step. On the other hand, Tab. VI indicates that the error prefactor α of Morales *et al.*'s formula is smaller than that of Yoshida's. Since such a difference in α can affect the estimated number of product-formula steps r required to achieve a fixed target error ε_{E}, it is plausible that the reduction in α outweighs the increase in the per-step decomposition count, leading to a smaller total gate count.

Fig. 1 also shows that Morales *et al.*'s tenth-order formula does not achieve the smallest total gate count in the considered regime. In general, higher-order formulae tend to involve more parameters to suppress higher-order error terms when approximating the time-evolution operator, and thus the per-step decomposition count m can become larger. Consequently, the per-step cost entering Eq. (6) may increase. For example, between the second-order formula and Morales *et al.*'s tenth-order formula with $m = 16$, the value from Eq. (6) differs by more than a factor of 50. When the target error is relatively loose, it is possible that the advantage of smaller error in higher-order formulae cannot compensate for this growth of the per-step cost, so that the difference in decomposition count cannot be overcome. Therefore, when an accuracy of only a few digits below chemical accuracy is sufficient, Morales *et al.*'s eighth-order formula may yield a smaller computational cost in terms of the total number of Pauli rotations.

We next consider the depth of R_Z-rotation layers as an alternative cost metric motivated by fault-tolerant compilation. Both the total gate count F and the depth of R_Z-rotation layers can be viewed as being influenced by the required number of product-formula steps r and a per-step overhead that may increase with the decomposition structure. However, the two metrics can respond differently to that per-step overhead. The total gate count F is given by summing the number of diagonal rotations

Model	R^2				
	2nd	4th	8th (Yoshida)	8th (Morales *et al.*)	10th (Morales *et al.*)
$\alpha = Cn^r$	0.9305	0.7178	0.1860	0.3640	0.5763
$\alpha = C(\log_2 n)^r$	0.8942	0.8250	0.4800	0.7282	0.8856
$\alpha = C(\log_2(\log_2 n))^r$	0.8049	0.8804	0.6492	0.9015	0.9192

TABLE I. R^2 values for each fitting model.

over the stages within a product-formula step, whereas the depth of R_Z-rotation layers depends on how those diagonal rotations can be parallelized after compiling each commuting group into Z-diagonal form.

As a result, a product formula that yields a smaller F does not necessarily yield a smaller depth of R_Z-rotation layers. In our results, Morales *et al.*'s eighth-order formula gives the smallest F over the range shown in Fig. 1, while Fig. 2 exhibits a crossover in the large-system extrapolation where the fourth-order formula becomes the lowest-depth choice. This behavior suggests that, for larger systems, the reduction in r associated with a smaller error prefactor can be offset by the growth of the per-step R_Z-layer depth, which may be governed by the conflict structure of the resulting Z and ZZ terms and the corresponding layerization. Therefore, when the cost is dominated by sequential rotation layers in fault-tolerant compilation, a lower-order product formula can be preferable even if it does not minimize the total number of rotations.

V.2. Construction of a low-coefficient, low-order product formula

We base our construction on the following two considerations:

1. When $\frac{e^{8/5}}{3} > \frac{\alpha}{\varepsilon_E} > 0$, reducing the coefficient α is more effective for lowering the total gate count than increasing the order p of the product formula.

2. Product formulae constructed with the methodology of Morales *et al.* exhibit smaller α compared to conventional formulae.

When ε_E is one order of magnitude larger than CA, the α of Morales *et al.*'s eighth-order formula, which minimizes the total gate count, satisfies $\frac{e^{8/5}}{3} > \frac{\alpha}{\varepsilon_E} > 0$ for many H-chains, as shown in Appendix H. Moreover, the α of Morales *et al.*'s formula is smaller than that of other formulae. Therefore, by constructing a product formula of lower order than eighth using Morales *et al.*'s methodology, one can further reduce the total gate count near CA compared to their eighth-order formula.

Therefore, we constructed a fourth-order product formula using Morales *et al.*'s methodology. The parameters of this formula, with $m = 2$, are

$$w_1 = 0.42008729$$
$$w_2 = 0.40899193$$

When constructing their eighth- and tenth-order formulae, Morales *et al.* optimized the sum of squared residuals between $e^{(A+B)t}$ and $S^{(m)}(t, w_1, ..., w_m)$ to be less than 10^{-22}, whereas in our work the optimized sum of squared residuals is on the order of 10^{-16}. We determined the parameters using the same methodology as Morales *et al.* Specifically, we expand both the exact $e^{(A+B)t}$ and $S^{(m)}(t, w_1, ..., w_m)$ constructed by Yoshida's fractal method into their Taylor series, compute the difference of coefficients for the same operators, square all such differences for terms below the desired order, and take the sum to obtain the squared residual. We then optimize the parameters w_i such that the squared residual falls below a threshold. In this study, the squared residual is on the order of 10^{-16}.

Fig. 3 compares the total gate count F at the target energy error $\varepsilon_{\mathrm{E}} = 1.6\times10^{-4}$ Ha for the eighth-order product formula by Morales *et al.* and our newly constructed fourth-order product formula, using the same evaluation protocol as in Fig. 1. We choose Morales *et al.*'s eighth-order formula as a reference because it yields the smallest total gate count F among the product formulae considered in our benchmark. For H_3 through H_{15}, the new fourth-order formula yields a smaller F than the eighth-order formula by Morales *et al.* The extrapolated scaling also favors the fourth-order formula, which attains a lower F up to 100 qubits; in particular, at 100 qubits, the new fourth-order formula reduces the total number of rotations by approximately 20% compared to Morales *et al.*'s eighth-order formula.

In addition to F, we also examine a compilation-aware metric, the depth of R_Z-rotation layers. Fig. 4 compares this depth between our new fourth-order formula and the original fourth-order formula at the same target error $\varepsilon_{\mathrm{E}} = 1.6 \times 10^{-4}$ Ha. We choose the original fourth-order formula as a baseline because it attains the smallest depth of R_Z-rotation layers at 100 qubits among the formulae considered in our benchmark. Over the evaluated H-chains, the new fourth-order formula consistently yields a smaller depth. The dashed curve in Fig. 4 shows the absolute difference in the estimated depths, indicating that the improvement persists across system sizes and becomes more pronounced in the extrapolated regime.

TABLE II. Comparison of R^2 between the conventional and new fourth-order formulae.

Model	R^2	
	4th	4th (new)
$\alpha = Cn^r$	0.7178	0.9063
$\alpha = C(\log_2 n)^r$	0.8250	0.9514
$\alpha = C(\log_2(\log_2 n))^r$	0.8804	0.9402

Moreover, at 100 qubits, the new fourth-order formula reduces the total number of rotations by approximately 30% compared to the original fourth-order formula.

The coefficients of determination R^2 of the new formula for the three fitting models described in Sec. IV.1 are listed in Tab. II. The new fourth-order formula achieves larger R^2 values than the conventional fourth-order formula.

From Tab. VI, the range of α for the new fourth-order formula is about 10^{-6} to 10^{-5}, whereas for the conventional fourth-order formula it is 10^{-4} to 10^{-2}. Thus, the new formula achieves smaller α and also exhibits slower growth of α.

VI. CONCLUSIONS AND OUTLOOK

In this study, we compared the computational cost of several product formulae for estimating molecular ground state energies using quantum phase estimation (QPE) with a prescribed target accuracy. As benchmark systems, we considered one-dimensional hydrogen chains from H_2 to H_{15}, and we primarily quantified the cost by the total number of Pauli rotations F required to achieve the target energy error.

For the evaluated H-chains, we did not find an accuracy regime in which Yoshida's eighth-order product formula minimizes F. Instead, the eighth-order formula constructed by Morales *et al.* consistently yields a smaller total rotation count. This suggests that, even for molecular Hamiltonians more complex than H-chains, employing the product-formula constructions of Morales *et al.* can be advantageous from a resource perspective when computing ground state energies via QPE. We also observed that, near chemical accuracy, increasing the order of the product formula does not necessarily reduce the overall cost once the full QPE workload is taken into account: although higher-order formulae can suppress the time-evolution approximation error, the increased decomposition structure can lead to a higher total cost.

Regarding the behavior of the error term, we observed a trend that the coefficient α in the fit $\delta E \approx \alpha t^p$ grows more slowly with the number of spin orbitals for product formulae constructed by the methodology of Morales *et al.* than for conventional constructions. Motivated by this observation, we constructed a new fourth-order product formula based on the same methodology. For H_3 through H_{15}, this new fourth-order formula yields a smaller F than Morales *et al.*'s eighth-order formula, and the extrapolated scaling indicates that the advantage persists to larger qubit counts. In particular, at 100 qubits, the new fourth-order formula reduces the total number of rotations by approximately 20% compared to Morales *et al.*'s eighth-order formula.

In addition to the total rotation count, we evaluated a compilation-aware metric, the depth of R_Z-rotation layers, motivated by fault-tolerant compilation where the number of sequential rotation layers can be a relevant proxy for latency. Since this metric depends on how the diagonal rotations can be parallelized after Z-diagonalizing each commuting group, a product formula that is favorable in terms of F does not necessarily remain favorable in terms of the R_Z-layer depth. Indeed, in the large-system extrapolation, we observed cases where the ranking by F and the ranking by the R_Z-layer depth do not coincide, suggesting that the preferred product formula can depend on the chosen cost metric. We also found that our new fourth-order formula improves the R_Z-layer depth relative to the original fourth-order formula, indicating that the proposed construction can be beneficial from both rotation-count and depth-based perspectives.

In this work we simulated molecules up to H_{15}. It will therefore be important to investigate, for larger systems, how the product-formula error term varies with the number of spin orbitals. Moreover, comparisons employing different product formulae and alternative QPE algorithms will also be necessary. Furthermore, to further reduce the number of Pauli rotations and the depth of R_Z-rotation layers, it is promising to incorporate Hamiltonian preprocessing techniques such as double factorization and tensor hypercontraction, which can reduce the number of Hamiltonian terms and the sum of coefficient magnitudes.

ACKNOWLEDGMENTS

This work is supported by JST COI-NEXT Grant No. JPMJPF2014. K.M. is supported by JST FOREST Grant No. JPMJFR232Z, JSPS KAKENHI Grant No. 23H03819, 24K16980 and JST CREST Grant No. JPMJCR24I4.

[1] A. Y. Kitaev, Quantum measurements and the abelian stabilizer problem, arXiv:quant-ph/9511026 (1995), arXiv:quant-ph/9511026 [quant-ph].

[2] D. S. Abrams and S. Lloyd, Physical Review Letters **83**,

5162 (1999).

[3] H. F. Trotter, Proceedings of the American Mathematical Society **10**, 545 (1959).

[4] M. Suzuki, Communications in Mathematical Physics **51**, 183 (1976).

[5] G. H. Low and I. L. Chuang, Quantum **3**, 163 (2019).

[6] A. Gilyén, Y. R. Sanders, G. H. Low, and N. Wiebe, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC '19)* (ACM, 2019) pp. 193–204.

[7] S. Bravyi and A. Kitaev, Physical Review A **71**, 022316 (2005).

[8] C. Gidney, N. Shutty, and C. Jones, Magic state cultivation: Growing t states as cheap as CNOT gates, arXiv:2409.17595 (2024), arXiv:2409.17595 [quant-ph].

[9] Y. Hirano, T. Itogawa, and K. Fujii, in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)* (IEEE, 2024) p. 843–853.

[10] Y. Akahoshi, K. Maruyama, H. Oshima, S. Sato, and K. Fujii, PRX Quantum **5**, 010337 (2024).

[11] M. E. Morales, P. Costa, G. Pantaleoni, D. K. Burgarth, Y. R. Sanders, and D. W. Berry, arXiv preprint arXiv:2210.15817 (2022).

[12] H. Yoshida, Physics letters A **150**, 262 (1990).

[13] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Proceedings of the national academy of sciences **114**, 7555 (2017).

[14] M. Suzuki, Physics Letters A **146**, 319 (1990).

[15] S. Blanes and F. Casas, Linear algebra and its applications **378**, 135 (2004).

[16] W. Kahan and R.-C. Li, Mathematics of Computation **66**, 1089 (1997).

[17] Q. Sun, T. C. Berkelbach, N. S. Blunt, G. H. Booth, S. Guo, Z. Li, J. Liu, J. D. McClain, E. R. Sayfutyarova, S. Sharma, S. Wouters, and G. K.-L. Chan, Wiley Interdisciplinary Reviews: Computational Molecular Science **8**, e1340 (2018).

[18] Q. Sun, X. Zhang, S. Banerjee, P. Bao, M. Barbry, N. S. Blunt, N. A. Bogdanov, G. H. Booth, J. Chen, Z.-H. Cui, J. J. Eriksen, Y. Gao, S. Guo, J. Hermann, M. R. Hermes, K. Koh, P. Koval, S. Lehtola, Z. Li, J. Liu, N. Mardirossian, J. D. McClain, M. Motta, B. Mussard, H. Q. Pham, A. Pulkin, W. Purwanto, P. J. Robinson, E. Ronca, E. R. Sayfutyarova, M. Scheurer, H. F. Schurkus, J. E. T. Smith, C. Sun, S.-N. Sun, S. Upadhyay, L. K. Wagner, X. Wang, A. White, J. D. Whitfield, M. J. Williamson, S. Wouters, J. Yang, J. Yu, T. Zhu, T. C. Berkelbach, S. Sharma, A. Y. Sokolov, and G. K.-L. Chan, Journal of Chemical Physics **153**, 024109 (2020).

[19] G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, Physical Review A **64**, 022319 (2001).

[20] J. R. McClean, I. D. Kivlichan, K. J. Sung, D. Steudtner, Y. Cao, C. Dai, E. S. Fried, C. Gidney, B. Gimby, T. Häner, J. Izaac, Z. Jiang, M. Kusumoto, Y. Li, S. McArdle, M. Neeley, T. E. O'Brien, B. O'Gorman, M. Ozols, A. D. Patterson, J. Romero, N. C. Rubin, N. P. D. Sawaya, K. Setia, S. Sim, M. Steudtner, Q. Sun, W. Sun, D. Wang, F. Zhang, P. D. Nation, A. Aspuru-Guzik, P. J. Love, A. Mezzacapo, and R. Babbush, Openfermion: The electronic structure package for quantum computers (2017), arXiv:1710.07629 [quant-ph].

[21] J. R. McClean, N. C. Rubin, K. J. Sung, I. D. Kivlichan, X. Bonet-Monroig, Y. Cao, C. Dai, E. S. Fried, C. Gidney, B. Gimby, P. Gokhale, T. Häner, T. Hardikar, V. Havlíček, O. Higgott, C. Huang, J. Izaac, Z. Jiang, X. Liu, S. McArdle, M. Neeley, T. E. O'Brien, B. O'Gorman, M. Ozols, D. Packwood, R. Santagati, M. S. Scheurer, K. Setia, S. Sim, S. Singh, V. Smelyanskiy, M. Steudtner, Q. Sun, W. Sun, D. Wang, F. Zhang, Y. Zhang, P. D. Nation, A. Aspuru-Guzik, P. J. Love, A. Mezzacapo, and R. Babbush, Quantum Science and Technology **5**, 034014 (2020).

[22] W. Inoue, K. Aoyama, Y. Teranishi, K. Kanno, Y. O. Nakagawa, and K. Mitarai, Phys. Rev. Res. **6**, 013096 (2024).

Appendix A: Estimation of β

We describe how to obtain the constant β. We assume the quantum phase estimation (QPE) based on the quantum Fourier transform. Let T denote the total number of applications of controlled-U. Let ϕ be the true phase, $\hat{\phi}(T)$ the estimated phase at a given T, and $\varepsilon(T)$ the phase error. In QFT-based QPE, the probability of obtaining outcome j is, with $\Delta_j = \theta_j - \phi$,

$$p(j \mid \phi, T) \;=\; \frac{1}{T^2}\,\frac{\sin^2\!\big(\frac{T}{2}\,\Delta_j\big)}{\sin^2\!\big(\frac{1}{2}\,\Delta_j\big)}, \qquad \sum_{j=0}^{T-1} p(j \mid \phi, T) = 1.$$

We sample measurement outcomes from this distribution and take the most frequent θ_j as the estimator $\hat{\phi}(T)$.

Given integers K, N, and N_{rep}, prepare $\{T_i\}_{i=1}^{K}$ with $T_i \in \mathbb{Z}$. For each $T = T_i$, perform one run as follows. First, draw N measurement samples from the above QFT–QPE distribution and take the most frequent discrete phase as the estimated phase $\hat{\phi}^{(r)}(T_i)$, where r indexes the run. Next, map both ϕ and $\hat{\phi}^{(r)}(T_i)$ into the interval $[-\pi, \pi]$ and define

$$\hat{\varepsilon}^{(r)}(T_i) \;=\; \big|\,\hat{\phi}^{(r)}(T_i) - \phi\,\big|.$$

Repeat this entire run independently N_{rep} times and, for the given ϕ, take the median of $\big\{\hat{\varepsilon}^{(r)}(T_i)\big\}_{r=1}^{N_{\text{rep}}}$ as $\varepsilon(T_i)$.

Plot $\varepsilon(T)$ versus T and fit the model $\log \varepsilon(T) = \log \beta - \log T$ to obtain β. Fig. 5 shows the plot, which follows $\varepsilon(T) \propto T^{-1}$. In this instance, we obtain $\beta = 1.43$. Since β depends on ϕ, we repeat the above procedure for 100 randomly sampled ϕ values and use their mean, $\beta = 1.20$, in this study.

Appendix B: Variation of the total gate count with respect to α and p

We define $f(\alpha, p)$ and $g(\alpha, p)$ as

$$f(\alpha, p) = \frac{1+p}{p\varepsilon_E}\left(\frac{\alpha(1+p)}{\varepsilon_E}\right)^{\frac{1}{p}},$$
$$g(\alpha, p) = [(4p+2)(J-1)+1] \times \beta\,.$$

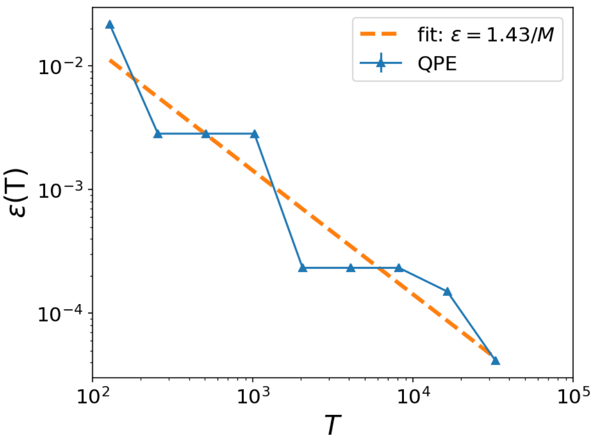


FIG. 5. Relationship between the error ε and the cost T in QFT-based QPE. Blue dots show the error of the estimator at each T. The orange dashed line is the fit to the model $\varepsilon = \beta/T$ with slope fixed to -1. $\phi = -1.5$, $N = 100$, $N_{\rm rep} = 10$

We have $F = f \times g$. Here we assume $m \approx p$. First, for $\partial F/\partial\alpha$, we obtain

$$\frac{\partial F}{\partial \alpha} = g * \frac{\partial f}{\alpha} = g * \left(\frac{1}{p}\alpha^{-1} f\right) = \frac{F}{p\alpha}\,. \tag{B1}$$

Next, for $\partial F/\partial p$, the following relation holds:

$$\frac{\partial(\ln f)}{\partial p} = \frac{1}{f}\frac{\partial f}{\partial p}\,. \tag{B2}$$

Therefore,

$$\frac{\partial(\ln f)}{\partial p} = -\frac{1}{p^2}\ln\frac{\alpha(1+p)}{\varepsilon_E}, \tag{B3}$$

$$\begin{aligned}\frac{\partial F}{\partial p} &= \left\{f\frac{\partial(\ln f)}{\partial p}\right\} g + f\,\{4(J-1)\cdot\beta\}\\ &= F\left\{-\frac{1}{p^2}\ln\frac{\alpha(1+p)}{\varepsilon_E} + \frac{4(J-1)}{(4p+2)(J-1)+1}\right\}.\end{aligned} \tag{B4}$$

For $\alpha \in [10^{-16}, 10^{-3}]$, $p \in [2, 10]$, $\varepsilon_E \in [10^{-5}, 10^{-3}]$, and $J \gg 1$, we find

$$\frac{\partial F}{\partial \alpha} = \frac{F}{10\cdot 10^{-3}} > 0\,.$$

On the other hand, considering Eq. (B4), when the coefficient α is smaller than the target error ε_E, i.e., $1 > \alpha/\varepsilon_E > 0$, we obtain

$$\begin{aligned}&\frac{\partial F}{\partial p} \sim F\frac{h(p)}{p^2},\\ &h(p) = -\ln\frac{\alpha(1+p)}{\varepsilon_E} + \frac{4p^2}{(4p+2)},\\ &\frac{dh(p)}{dp} = \frac{4p^3+4p^2-1}{(2p+1)^2(p+1)} > 0 \quad (p>2),\\ &h(p) > h(2) = -\ln\frac{3\alpha}{\varepsilon_E} + \frac{8}{5} > 0 \quad \left(\frac{e^{8/5}}{3} > \frac{\alpha}{\varepsilon_E} > 0\right).\end{aligned}$$

Thus,

$$\frac{\partial F}{\partial p} > 0\,.$$

Appendix C: Z-diagonalization of grouped Hamiltonian terms and R_Z layer depth estimation

This Appendix summarizes the Z-diagonalization rules and the procedure used to estimate the depth of R_Z-rotation layers for each commuting group H_i introduced in Sec. III.

1. Building blocks and structure of grouped terms

We define the fermionic number operator

$$n_p := a_p^\dagger a_p, \tag{C1}$$

and the Hermitian hopping operator

$$A_{pq} := a_p^\dagger a_q + a_q^\dagger a_p \qquad (p \neq q). \tag{C2}$$

In our implementation, the grouped Hamiltonian terms produced by the Inoue grouping are expressed as linear combinations of products of these building blocks, namely

$$\begin{aligned}&H_i = \sum_\nu \gamma_{i\nu}\, B_{i\nu}, \\ &B_{i\nu} \in \Big\{A_{pq},\ n_p,\ A_{pq}n_r,\ A_{pq}A_{rs},\ n_pn_q,\ \dots\Big\},\end{aligned} \tag{C3}$$

where the set of monomials $B_{i\nu}$ is closed under taking products appearing in the grouped representation.

2. Z-diagonalization rule for A_{pq}

For each hopping operator A_{pq}, we apply a basis change (implemented by a Clifford circuit in our cost model) that diagonalizes it in the fermionic occupation basis as

$$A_{pq} \;\mapsto\; n_p - n_q. \tag{C4}$$

As a consequence, any monomial built from A_{pq} and n_r is mapped to a polynomial in number operators.

3. Jordan-Wigner mapping to Z-type operators

Under the Jordan-Wigner (JW) transformation, the number operator becomes a single-qubit Z operator,

$$n_p \mapsto \frac{1-Z_p}{2}. \tag{C5}$$

Combining Eqs. (C4) and (C5), we obtain the following conversion rules used throughout our depth estimation:

$$\begin{aligned} A_{pq} &\mapsto n_p - n_q \mapsto \frac{-Z_p+Z_q}{2}, &\text{(C6)}\\ A_{pq}n_r &\mapsto (n_p-n_q)n_r \mapsto \frac{-Z_p+Z_q}{2}\cdot\frac{1-Z_r}{2}, &\text{(C7)}\\ A_{pq}A_{rs} &\mapsto (n_p-n_q)(n_r-n_s)\\ &\mapsto \frac{-Z_p+Z_q}{2}\cdot\frac{-Z_r+Z_s}{2}. &\text{(C8)}\end{aligned}$$

Expanding the products shows that each mapped monomial becomes a linear combination of Z-type operators. In particular, for the monomials considered in this work, the expansion contains only constant terms, single-qubit Z terms, and two-qubit ZZ terms.

4. From Z-diagonal form to R_Z-layer depth

For each commuting group H_i, we assume a common Clifford basis-change circuit U_i that simultaneously diagonalizes all terms in the group, and we write

$$U_i^\dagger H_i U_i = \sum_\ell c_{i\ell}\, Z_{i\ell}, \tag{C9}$$

$$e^{-itH_i} = U_i \exp\left(-it\sum_\ell c_{i\ell}\, Z_{i\ell}\right) U_i^\dagger, \tag{C10}$$

where each $Z_{i\ell}$ is a (typically short) Z string. The diagonal part in Eq. (C10) is implemented using commuting rotations generated by the Z and ZZ terms appearing in Eq. (C9).

We define the depth of R_Z-rotation layers as the minimum number of sequential layers required to apply all such rotations when rotations acting on disjoint sets of qubits are executed in parallel. To estimate this layer depth, we construct a *conflict graph* whose vertices correspond to the individual Z-type rotation terms in Eq. (C9), and we connect two vertices if the corresponding rotations act on at least one common qubit. The required number of R_Z layers is then estimated by a graph-coloring procedure on this conflict graph. In this estimate, we neglect the depth contribution from the Clifford basis-change circuit U_i and only count the layers of the diagonal Z-type rotations.

Appendix D: Eighth- and tenth-order product formulae

The parameters of the eighth- and tenth-order product formulae used in this study are listed in Tab. III and Tab. IV.

Appendix E: Total computational cost for achieving the target error

We first set the condition for accuracy. Let the error from phase estimation be $\varepsilon_{\mathrm{QPE}}$ and the error from the product formula be αt^p with $\alpha, p \in \mathbb{R}$. To achieve the desired target error ε_{E}, it is necessary to satisfy

$$\varepsilon_{\mathrm{QPE}} + \alpha t^p \le \varepsilon_{\mathrm{E}}\,. \tag{E1}$$

Next, we consider the computational cost. The decomposition count of the product formula is independent of accuracy, as given by Eq. (6). Thus, the computational cost depends on the precision of phase estimation, according to Eq. (1). Since the phase of the time-evolution operator is estimated as Et, the parameter ε in Eq. (1) is replaced by $\varepsilon_{\mathrm{QPE}}t$. Therefore, minimizing the cost for achieving the target error ε_{E} reduces to the following optimization problem:

$$\text{minimize} \quad \frac{\beta}{\varepsilon_{\mathrm{QPE}}t} \quad \text{subject to} \quad \varepsilon_{\mathrm{QPE}} + \alpha t^p \le \varepsilon_{\mathrm{E}}\,. \tag{E2}$$

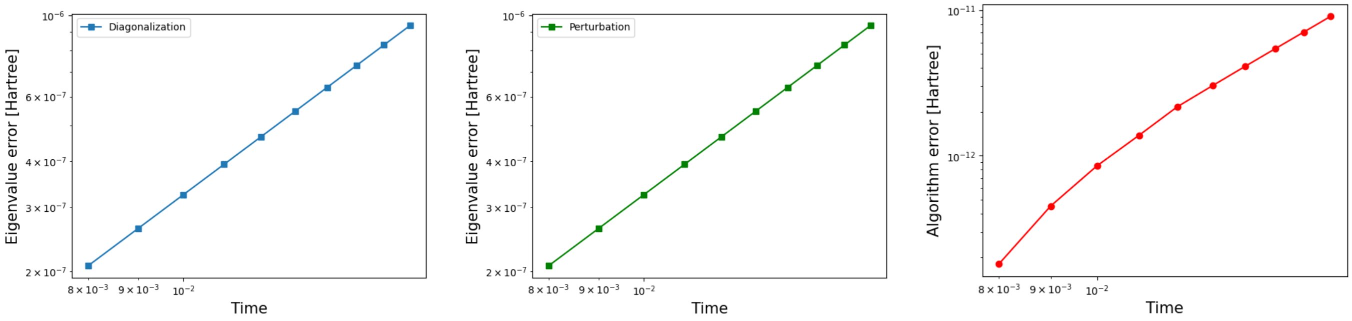
Solving this problem using the Lagrange multiplier method, we find that the minimum lies on the boundary of the constraint and is given by

$$t = \left(\frac{\varepsilon_{\mathrm{E}}}{\alpha(p+1)}\right)^{\frac{1}{p}}, \qquad \varepsilon_{\mathrm{QPE}} = \varepsilon_{\mathrm{E}}\left(\frac{p}{p+1}\right), \tag{E3}$$

$$\text{minimized} \quad \frac{\beta}{\varepsilon_{\mathrm{QPE}}t} = \beta\left(\frac{1+p}{p\varepsilon_{\mathrm{E}}}\right)\left(\frac{\alpha(1+p)}{\varepsilon_{\mathrm{E}}}\right)^{\frac{1}{p}}. \tag{E4}$$

Here, the value given by Eq. (1) represents the number of U gates applied in phase estimation. When using a product formula, each U gate consists of the number of exponentials given by Eq. (6). Therefore, the minimum gate count required to achieve the target error is expressed as Eq. (8):

$$\left\{\beta\left(\frac{1+p}{p\varepsilon_E}\right)\left(\frac{\alpha(1+p)}{\varepsilon_E}\right)^{\frac{1}{p}}\right\} \times \{(4m+2)(J-1)+1\}\,. \tag{E5}$$



(a) (a) Eigenvalue error computed by direct diagonalization

(b) (b) Eigenvalue error computed by a perturbative method

(c) (c) Errors from diagonalization and from the perturbative method

FIG. 6. Comparison of eigenvalue errors computed by direct diagonalization and by the perturbative method

Appendix F: Error evaluation by perturbation theory

Let $(E_0, |\psi_0\rangle)$ be the ground state energy and ground state of the Hamiltonian H. Let $U_{\mathrm{pf}}(t)$ be a pth-order product-formula approximation of the time-evolution operator. We can write $U_{\mathrm{pf}}(t)$ as

$$U_{\mathrm{pf}}(t) = e^{-iH_{\mathrm{eff}}t},$$

where

$$H_{\mathrm{eff}} = H + V \tag{F1}$$

for some operator V satisfying $\|V\| = O\big(t^{\,p}\big)$. Let E_{pf} denote the ground state energy of $H_{\mathrm{eff}}(t)$, and define

$$\delta E := E_{\mathrm{pf}} - E_0.$$

We define the deviation between the product-formula evolution and the exact evolution as

$$|\psi'(t)\rangle := U_{\mathrm{pf}}(t)\,|\psi_0\rangle\,, \tag{F2}$$

$$|\Delta\psi(t)\rangle := |\psi'(t)\rangle - e^{-iE_0t}\,|\psi_0\rangle\,. \tag{F3}$$

By first-order perturbation theory for $H_{\mathrm{eff}}(t) = H+V$, the ground state energy can be approximated as

$$E_{\mathrm{pf}} = E_0 + \langle\psi_0|\,V\,|\psi_0\rangle + O(\|V\|^2).$$

Hence

$$\delta E = \langle\psi_0|\,V\,|\psi_0\rangle + O(\|V\|^2), \tag{F4}$$

and substituting $\|V\| = O(t^p)$ from Eq. (F1) gives

$$\delta E = \langle\psi_0|\,V\,|\psi_0\rangle + O\big(t^{\,2p}\big).$$

We are therefore interested in calculating $\langle\psi_0|\,V\,|\psi_0\rangle$. From Eq. (F1),

$$\begin{aligned} U_{\mathrm{pf}}(t) &= e^{-i(H+V)t} \\ &= e^{-iHt}\Big(\mathbb{I} - itV + O(t^2\|V\|)\Big) \\ &= e^{-iHt}\Big(\mathbb{I} - itV + O(t^{p+2})\Big). \end{aligned}$$

We therefore have,

$$|\psi'(t)\rangle = e^{-iE_0t}\Big(\,|\psi_0\rangle - it\,V\,|\psi_0\rangle\Big) + O(t^{p+2}),$$

and thus,

$$|\Delta\psi(t)\rangle = -ite^{-iE_0t}V\,|\psi_0\rangle + O(t^{p+2}).$$

We can take inner product between $|\Delta\psi(t)\rangle$ and $|\psi_0\rangle$ to obtain

$$\begin{aligned} \langle\psi_0|\Delta\psi(t)\rangle &= -ite^{-iE_0t}\,\langle\psi_0|\,V\,|\psi_0\rangle + O(t^{p+2}) \\ &= -ite^{-iE_0t}\,\big(\delta E + O(t^{2p})\big) + O(t^{p+2}). \end{aligned} \tag{F5}$$

Using $e^{-iE_0t} = \cos(E_0t) - i\sin(E_0t)$ and taking the real part of Eq. (F5), we obtain

$$\Re\,\langle\psi_0|\Delta\psi(t)\rangle = t\,\delta E\,\sin(E_0t) + O\big(t^{2p+1}\big) + O(t^{p+2}).$$

Assuming $\sin(E_0t) \neq 0$, we conclude

$$\delta E = \frac{\Re\,\langle\psi_0|\Delta\psi(t)\rangle}{t\,\sin(E_0t)} + O\big(t^{\,p+1}\big), \tag{F6}$$

which is Eq. (F6) in the small-t limit.

We compare the eigenvalue error of the ground state obtained by the perturbative method in Sec. 2 with that from direct diagonalization. Figs. 6a and 6b show, for the H_2 Hamiltonian with the time-evolution operator computed by a second-order product formula, the eigenvalue error evaluated by direct diagonalization and by the perturbative method, respectively. Fig. 6c plots the discrepancy between the two evaluations, hereafter referred to as the algorithmic error. As Fig. 6c indicates, a nonzero algorithmic error exists but it remains much smaller than the eigenvalue error over the range studied. We therefore estimate the total gate count based on the ground-state eigenvalue error obtained via the perturbative method.

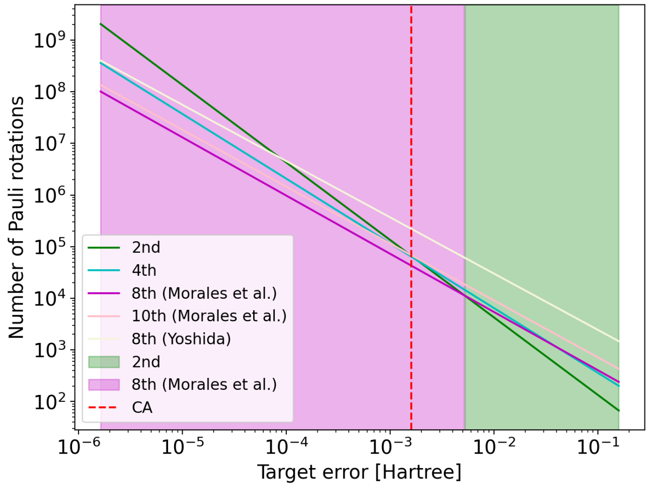


FIG. 7. Number of Pauli rotations required to reach a target error for the H_2 molecule. Shaded regions indicate which product formula attains the minimum count across each error regime. The red dashed line denotes chemical accuracy (CA).

Appendix G: Estimation of the total gate count for H_2

Fig. 6c shows how the error of the time-evolution operator for H_2, computed using product formulae, varies as a function of t. We determine α and p by fitting this log-log plot with a linear function of t. Appendix H lists the values of α and p for each H-chain.

Using these α and p, we evaluate Eq. (E4) to obtain the total gate count. We repeat the same procedure for the other product formulas. Fig. 7 plots Eq. (E4) as a function of ε_E [Hartree]. Because ε_E denotes the desired target error, moving from right to left on the horizontal axis corresponds to lower error. The color bands in Fig. 7 indicate which product formula minimizes the total gate count. On the right side, up to $\varepsilon_E \approx 10^{-3}$, the second-order product formula yields the lowest total gate count. For $\varepsilon_E \lesssim 10^{-3}$, the eighth-order formula by Morales *et al.* achieves the minimum. Fig. 8 presents the same evaluation across H-chains in terms of the target error.

Appendix H: α, p for each H-chain

Tab. VI lists the coefficients α obtained by fitting with p fixed to the order of each product formula. This setting makes it easier to compare the magnitude of the error terms among formulae of the same order.

TABLE III. Fourth-order product formula and Yoshida's eighth-order product formula

	4th order	Yoshida 8th order
w_1	-1.70241438391931	-1.61582374150097
w_2	1.35120719195965	-2.44699182370524
w_3	-	-0.0071698941970812
w_4	-	2.44002732616735
w_5	-	0.157739928123617
w_6	-	1.82020630970714
w_7	-	1.04242620869991

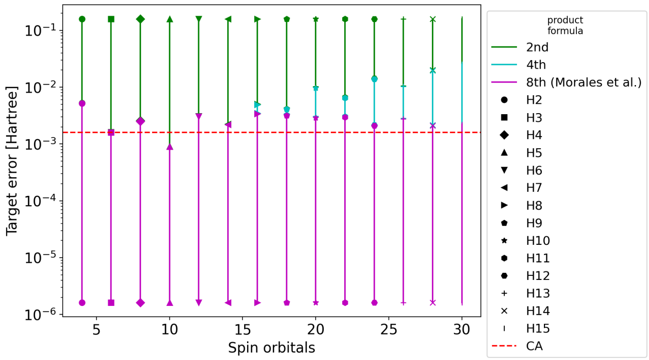


FIG. 8. For each target error, the product formula achieving the lowest total gate count is identified as a function of the number of spin orbitals in one-dimensional hydrogen chains. For each hydrogen chain (H_2–H_{15}), vertical lines indicate the error ranges, with the markers showing the endpoints. Colors indicate the ranges of target errors for which the corresponding product formula is optimal (lowest total gate count): green (second order), cyan (fourth order), magenta (eighth order). The red dashed line represents chemical accuracy (CA), $1\,\mathrm{kcal/mol} \approx 1.6 \times 10^{-3}\,\mathrm{Ha}$.

Appendix I: Best Product Formula for Any Target Error

For each H-chain, we identify, as a function of ε_E, the product formula that minimizes the total gate count. The results are shown in Fig.8. For H_2 to H_7, the second-order product formula yielded the lowest total gate count in the high-error region near CA. In contrast, for H8 to H15, the fourth-order product formula yielded the lowest total gate count in the high-error region near CA. In the lower-error regime beyond CA, the eighth-order formula newly constructed by Morales *et al.* achieved the smallest

TABLE IV. Morales *et al.*'s eighth- and tenth-order product formulae

	Morales 8th order	Morales 10th order
w_1	0.29137384767986663096528500968049	-0.4945013179955571856347147977644
w_2	0.26020394234904150277316667709864	0.2904317222970121479878414292093
w_3	0.18669648149540687549831902999911	0.34781541068705330937913890281003
w_4	-0.40049110428180105319963667975074	-0.98828132118546184603769781410676
w_5	0.15982762208609923217390166127256	0.98855187532756405235733957305613
w_6	-0.38400573301491401473462588779099	-0.34622976933123177430694714630668
w_7	0.56148845266356446893590729572808	0.20218952619073117554714280367018
w_8	-	0.13064273069786247787208895471461
w_9	-	-0.26441199183146805554735845490359
w_{10}	-	0.060999140559210408869096992291531
w_{11}	-	-0.6855442489606141359108973267028
w_{12}	-	-0.15843692473786584550599206557006
w_{13}	-	0.15414691779958299150286452215575
w_{14}	-	0.66715205827214320371061839297055
w_{15}	-	0.20411874474696598289603677693511
w_{16}	-	0.081207318210272593225087711441684

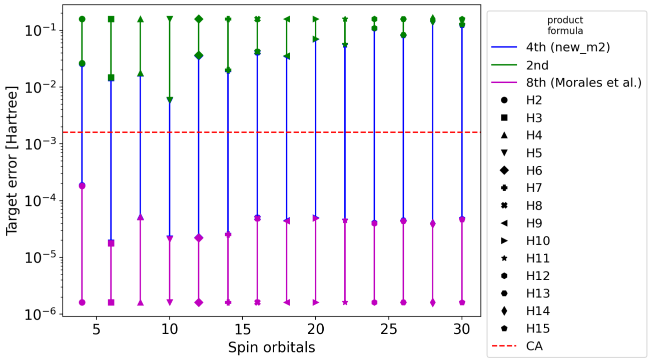


FIG. 9. The specifications of this figure are the same as Fig. 8. Colors indicate the error ranges where the corresponding product formula is optimal (lowest total gate count): green (second order), red (new fourth order), magenta (eighth order). The red dashed line represents chemical accuracy (CA), 1 kcal/mol ≈ 1.6×10^{-3} Ha.

gate count.

Fig. 9 shows the product formulae that yield the lowest total gate count for each target error, including this newly constructed formula. The new fourth-order formula, denoted 4th (new), yields the smallest total gate count near CA for all hydrogen chains from H_2 to H_{15}.

TABLE V. The product formula for each H-chain: α, p

H-chain		2nd	4th	8th (Yoshida)	8th (Morales)	10th (Morales)	4th (new)
H_2	α	0.00324305	0.00094925	0.00135268	6.51357383×10^{-10}	1.12229748×10^{-10}	4.6272997×10^{-6}
	p	2.00012893	3.95210826	11.21373142	7.88872563	9.81116352	3.998977
H_3	α	0.00494304	0.00204196	0.0224786	8.432258×10^{-9}	1.12229748×10^{-10}	9.78671221×10^{-6}
	p	2.00026564	3.96984987	10.43096998	7.94434778	9.81116352	3.99377781
H_4	α	0.01126311	0.00438536	0.07160648	2.64745466×10^{-8}	5.74622931×10^{-10}	2.13216796×10^{-5}
	p	2.00025898	3.95626609	10.05163505	7.94122101	9.67024335	3.98909614
H_5	α	0.00546397	0.00276758	0.06195283	2.51576155×10^{-8}	3.9145705×10^{-10}	1.26750251×10^{-5}
	p	2.00034615	3.95559043	9.96910435	7.91618746	9.69597769	3.95986405
H_6	α	0.01978758	0.0063653	0.12738943	9.22250417×10^{-8}	2.97549417×10^{-9}	3.0377915×10^{-5}
	p	2.00023256	3.92691346	9.19314467	7.98505758	9.44353262	3.95776299
H_7	α	0.01429684	0.0054076	0.10665487	6.61307591×10^{-8}	1.69714581×10^{-9}	2.3396525×10^{-5}
	p	2.00028403	3.9310129	9.3696503	7.92207769	9.54470209	3.91202941
H_8	α	0.02925554	0.00782246	0.042627	1.26509606×10^{-7}	4.82991411×10^{-9}	3.4148715×10^{-5}
	p	2.00012141	3.88414672	7.90464113	8.02985595	9.50736057	3.88102368
H_9	α	0.02513963	0.00708148	0.70088269	1.05829901×10^{-7}	3.98708458×10^{-9}	2.46957308×10^{-5}
	p	2.00015292	3.92357858	10.29089992	8.01326328	9.4065737	3.79313523
H_{10}	α	0.04223158	0.00701246	0.95275448	1.59891925×10^{-7}	7.81286665×10^{-9}	2.47452318×10^{-5}
	p	2.03385155	3.63733823	10.1058147	8.07154386	9.43618696	3.672347
H_{11}	α	0.03470004	0.00836608	0.71974725	1.32733903×10^{-7}	6.89937602×10^{-9}	1.57928922×10^{-5}
	p	2.00018902	3.93445927	10.05508955	8.19358341	9.29723326	3.51185744
H_{12}	α	0.05633722	0.0063569	0.26612231	1.69235704×10^{-7}	1.16791754×10^{-8}	1.35673507×10^{-5}
	p	2.17974829	3.54509946	9.13581948	8.35981579	9.38549927	3.38197958
H_{13}	α	0.04466339	0.00860154	0.00246769	1.75959803×10^{-7}	9.6648559×10^{-9}	9.18754984×10^{-6}
	p	2.00024639	3.88806929	6.28906563	8.07857139	9.37635348	3.20705395
H_{14}	α	0.06624539	0.00828989	17.30290053	2.07370696×10^{-7}	3.54311152×10^{-8}	6.41101449×10^{-6}
	p	2.0901986	3.83656559	11.55488901	8.2343376	7.30716629	3.01087982
H_{15}	α	0.05763799	0.00595476	9.56437897	1.83833925×10^{-7}	-	4.3472737×10^{-6}
	p	2.00037522	3.44766463	10.76698871	8.20404767	-	2.82313263

TABLE VI. α when fitting with fixed order p

H-chain	2nd	4th	8th (Yoshida)	8th (Morales)	10th (Morales)	4th (new)
H_2	0.0032416	0.00097860	8.9406×10^{-5}	6.2125×10^{-10}	4.6959×10^{-12}	4.6378×10^{-6}
H_3	0.0049387	0.0021096	0.0015095	8.2350×10^{-9}	1.0344×10^{-10}	9.9233×10^{-6}
H_4	0.011253	0.0046096	0.0068950	2.5821×10^{-8}	4.9839×10^{-10}	2.1846×10^{-5}
H_5	0.0054578	0.0029038	0.0065544	2.4276×10^{-8}	3.4331×10^{-10}	1.3861×10^{-5}
H_6	0.019772	0.0068892	0.032661	9.1640×10^{-8}	2.3401×10^{-9}	3.3376×10^{-5}
H_7	0.014283	0.022358	0.0058268	6.3337×10^{-8}	1.4421×10^{-9}	2.8464×10^{-5}
H_8	0.029242	0.0088674	0.047525	1.2861×10^{-7}	4.0498×10^{-9}	4.4517×10^{-5}
H_9	0.025125	0.025125	0.021699	1.0660×10^{-7}	3.0980×10^{-9}	3.9160×10^{-5}
H_{10}	0.041279	0.0088908	0.039058	1.6483×10^{-7}	6.1475×10^{-9}	5.2854×10^{-5}
H_{11}	0.034675	0.0090950	0.031866	1.4393×10^{-7}	5.1173×10^{-9}	4.8920×10^{-5}
H_{12}	0.051841	0.00856117	0.0475146	2.0612×10^{-7}	9.0664×10^{-9}	5.6776×10^{-5}
H_{13}	0.044622	0.0099206	0.033068	1.8184×10^{-7}	7.4745×10^{-9}	5.7654×10^{-5}
H_{14}	0.062448	0.010209	0.078746	2.3547×10^{-7}	1.1680×10^{-8}	6.3371×10^{-5}
H_{15}	0.057550	0.0085475	0.070054	1.9996×10^{-7}	-	6.63801×10^{-5}