Measurement-efficient quantum Krylov subspace diagonalisation

Zongkang Zhang, Anbang Wang, Xiaosi Xu, and Ying Li

Graduate School of China Academy of Engineering Physics, Beijing 100193, China

arXiv:2301.13353v3 [quant-ph] 8 Aug 2024

The Krylov subspace methods, being one category of the most important classical numerical methods for linear algebra problems, can be much more powerful when generalised to quantum computing. However, quantum Krylov subspace algorithms are prone to errors due to inevitable statistical fluctuations in quantum measurements. To address this problem, we develop a general theoretical framework to analyse the statistical error and measurement cost. Based on the framework, we propose a quantum algorithm to construct the Hamiltonian-power Krylov subspace that can minimise the measurement cost. In our algorithm, the product of power and Gaussian functions of the Hamiltonian is expressed as an integral of the real-time evolution, such that it can be evaluated on a quantum computer. We compare our algorithm with other established quantum Krylov subspace algorithms in solving two prominent examples. To achieve an error comparable to that of the classical Lanczos algorithm at the same subspace dimension, our algorithm typically requires orders of magnitude fewer measurements than others. Such an improvement can be attributed to the reduced cost of composing projectors onto the ground state. These results show that our algorithm is exceptionally robust to statistical fluctuations and promising for practical applications.

1 Introduction

Finding the ground-state energy of a quantum system is of vital importance in many fields of physics [1–3]. The Lanczos algorithm [4, 5] is one of the most widely used algorithms to solve this problem. It belongs to the Krylov subspace methods [6], in which the solution usually converges to the true answer with an increasing subspace dimension. However, such methods are unscalable for many-body systems because of the exponentially-growing Hilbert space dimension [7]. In quantum computing, there is a family of hybrid quantum-classical algorithms that can be regarded as a quantum generalisation of the classical Lanczos algorithm [8–20]. Following Refs. [11, 15, 16], we call them quantum Krylov subspace diagonalisation (KSD). These algorithms are scalable with the system size by carrying out the classically-intractable vector and matrix arithmetic on the quantum computer. They possess potential advantages as the conventional quantum algorithm to solve the ground-state problem, quantum phase estimation, requires considerable quantum resources [21, 22], while the variational quantum eigensolver is limited by the ansatz and classical optimisation bottlenecks [23, 24].

However, Krylov subspace methods are often confronted with the obstacle that small errors can cause large deviations in the ground-state energy. This issue is rooted in the Krylov subspace that is spanned by an almost linearly dependent basis [25]. Contrary to classical computing, in which one can exponentially suppress rounding errors by increasing the number of bits, quantum computing is inherently subject to statistical error. Since statistical error decreases slowly with the measurement number M as $\propto 1/\sqrt{M}$, an extremely large M can be required to reach an acceptable error. In this case, although quantum KSD algorithms perform well in principle, the measurement cost has to be assessed and optimised for realistic implementations [16, 20].

In this work, we present a general and rigorous analysis of the measurement cost in quantum

Zongkang Zhang: These two authors contributed equally.
Anbang Wang: These two authors contributed equally.
Ying Li: yli@gscaep.ac.cn

KSD algorithms. Specifically, we obtain an upper bound formula of the measurement number that is applicable to all quantum KSD algorithms. Then, we propose an algorithm to construct the Hamiltonian-power Krylov subspace [6]. In our algorithm, we express the product of Hamiltonian power and a Gaussian function of Hamiltonian as an integral of real-time evolution. In this way, the statistical error decreases exponentially with the power, which makes our algorithm measurement-efficient. We benchmark quantum KSD algorithms by estimating their measurement costs in solving the anti-ferromagnetic Heisenberg model and the Hubbard model. Various lattices of each model are taken in the benchmarking. It is shown that the measurement cost in our algorithm is typically orders of magnitude fewer than other algorithms to reach an error comparable to the classical Lanczos algorithm. We also demonstrate the measurement efficiency of our algorithm by composing a ground-state projector.

2 Krylov subspace diagonalisation

First, we introduce the KSD algorithm and some relevant notations. The algorithm starts with a reference state $|\varphi\rangle$ (or a set of reference states [10, 11]; we focus on the single-reference case in this work). Then we generate a set of basis states

$$|\phi_k\rangle = f_k(H)|\varphi\rangle, \tag{1}$$

where H is the Hamiltonian, and $f_1, f_2, \ldots, f_d$ are linearly-independent $(d-1)$-degree polynomials (to generate a d-dimensional Krylov subspace). For example, it is conventional to take the power (P) function $f_k(H) = H^{k-1}$ in the Lanczos algorithm. These states span a subspace called the Krylov subspace. We compute the ground-state energy by solving the generalised eigenvalue problem

$$\mathbf{Ha} = E\mathbf{Sa}, \tag{2}$$

where $\mathbf{H}_{k,q} = \langle\phi_k|H|\phi_q\rangle$ and $\mathbf{S}_{k,q} = \langle\phi_k|\phi_q\rangle$. Let E_{min} be the minimum eigenvalue of the generalised eigenvalue problem. The error in the ground-state energy is

$$\epsilon_K = E_{min} - E_g, \tag{3}$$

where E_g is the true ground-state energy. We call ϵ_K the subspace error. One can also construct generalised Krylov subspaces in which f_k are functions of H other than polynomials; see Table 1. Appendix A provides a more detailed introduction to the KSD algorithm.

In the literature, subspaces generated by variational quantum circuits [26–28] and stochastic time evolution [29] are also proposed. Quantum KSD techniques can also be used to mitigate errors caused by imperfect gates [30–32]. In this work, we focus on Hamiltonian functions because of their similarities to conventional Krylov subspace methods.

Abbr.	$f_k(x)$	Refs.
P	x^{k-1}	[19, 12]
CP	$T_{k-1}(x/h_{tot})$	[20]
GP	$x^{k-1}e^{-\frac{1}{2}x^2\tau^2}$	This work
IP	$x^{-(k-1)}$	[18]
ITE	$e^{-\tau(k-1)x}$	[8, 9]
RTE	$e^{-ix\Delta t\left(k-\frac{d+1}{2}\right)}$	[10–17]
F	$L^{-1}\sum_{l=1}^{L} e^{-i[x-\Delta E(k-1)]\Delta t\left(l-\frac{L+1}{2}\right)}$	[15]

Table 1: Operators $f_k(x)$ generating the basis of a (generalised) Krylov subspace. Here $x = H - E_0$, where E_0 is a constant. In different algorithms, f_k can be power (P), Chebyshev polynomial (CP), Gaussian-power (GP), inverse power (IP) or exponential [i.e. imaginary-time evolution (ITE), real-time evolution (RTE) and filter (F)] functions of the Hamiltonian. T_n is the n-th Chebyshev polynomial of the first kind, and $E_0 = 0$ in the CP basis. For a Hamiltonian expressed in the form $H = \sum_j h_j\sigma_j$, where σ_j are Pauli operators, $h_{tot} = \sum_j |h_j|$ is the 1-norm of coefficients. τ, Δt and ΔE are some real parameters.

3 Statistical error and regularisation

In addition to ϵ_K, the other error source is the statistical error. In quantum KSD algorithms, matrices $\mathbf{H}$ and $\mathbf{S}$ are obtained by measuring qubits at the end of certain quantum circuits. Measurements yield estimators $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$ of the two matrices, respectively. Errors in $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$ depend on the measurement number. Suppose the budget for each complex matrix entry is $2M$ measurements (the budget for each part is M). Variances of matrix entries have upper bounds in the form

$$\mathrm{Var}(\hat{\mathbf{H}}_{k,q}) \leqslant 2C_{\mathbf{H}}^2/M \tag{4}$$

and

$$\mathrm{Var}(\hat{\mathbf{S}}_{k,q}) \leqslant 2C_{\mathbf{S}}^2/M, \tag{5}$$

where $C_{\mathbf{H}}$ and $C_{\mathbf{S}}$ are some factors depending on the measurement protocol. For example, suppose each entry of $\mathbf{H}$ and $\mathbf{S}$ can be expressed in the form $\sum_s q_s\langle\varphi|U_s|\varphi\rangle$, where U_s are unitary operators, and we measure each term using the Hadamard test [33], then the variance of the entry has the upper bound $2C^2/M$, where $C = \sum_s |q_s|$.

We quantify the error in a matrix with the spectral norm. The distributions of $\|\hat{\mathbf{H}} - \mathbf{H}\|_2$ and $\|\hat{\mathbf{S}} - \mathbf{S}\|_2$ depend on not only the measurement number but also the correlations between matrix entries. We can measure matrix entries independently, then their distributions are uncorrelated. However, certain matrix entries take the same value, and we can measure them collectively. For example, for the P basis, all entries $\mathbf{H}_{k-q,q}$ take the same value as $\mathbf{H}_{k-1,1}$. We only need to measure one of them on the quantum computer, and then these entries become correlated. In Appendix B, we analyse the distributions of spectral norms for each basis in Table 1.

Errors in matrices cause an error in the ground-state energy in addition to ϵ_K. It is common that the subspace basis is nearly linearly dependent, which makes the overlap matrix $\mathbf{S}$ ill-conditioned. Then, even small errors in matrices may cause a serious error in the ground-state energy. Thresholding is a method to overcome this problem [8, 16]. In this work, we consider the regularisation method; see Algorithm 1. An advantage of the regularisation method is that by taking a proper regularisation parameter η, the resulting energy $\hat{E}_{min}$ is variational, i.e. $\hat{E}_{min} \geqslant E_g$ up to a controllable failure probability.

We propose to choose the regularisation parameter η according to the distributions of $\|\hat{\mathbf{H}} - \mathbf{H}\|_2$ and $\|\hat{\mathbf{S}} - \mathbf{S}\|_2$. Let κ be the permissible failure probability. We take η such that

$$\begin{aligned} &\Pr\left(\|\hat{\mathbf{H}} - \mathbf{H}\|_2 \leqslant C_{\mathbf{H}}\eta \text{ and } \|\hat{\mathbf{S}} - \mathbf{S}\|_2 \leqslant C_{\mathbf{S}}\eta\right) \\ \geqslant\ & 1 - \kappa. \end{aligned} \tag{6}$$

In Table 2, we list η satisfying the above condition for each basis. The value of η depends on the failure probability κ and the measurement number M. Taking η in this way, an upper bound of the energy error is given in the following lemma.

Algorithm 1 Regularised quantum Krylov-subspace diagonalisation algorithm.

1: Input $\hat{\mathbf{H}}$, $\hat{\mathbf{S}}$, $C_{\mathbf{H}}$, $C_{\mathbf{S}}$ and $\eta > 0$.
2: Solve the generalised eigenvalue problem

$$(\hat{\mathbf{H}} + C_{\mathbf{H}}\eta)\mathbf{a} = E(\hat{\mathbf{S}} + C_{\mathbf{S}}\eta)\mathbf{a}$$

3: Output the minimum eigenvalue $\hat{E}_{min}$.

Lemma 1. *Let*

$$E'(\eta, \mathbf{a}) = \frac{\mathbf{a}^\dagger(\mathbf{H} + 2C_{\mathbf{H}}\eta)\mathbf{a}}{\mathbf{a}^\dagger(\mathbf{S} + 2C_{\mathbf{S}}\eta)\mathbf{a}}. \tag{7}$$

Under conditions $E_g < 0$ and $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) < 0$, the following inequality holds,

$$\Pr\left(E_g \leqslant \hat{E}_{min} \leqslant \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a})\right) \geqslant 1 - \kappa. \tag{8}$$

See Appendix C for the proof. Notice that we can always subtract a sufficiently large positive constant from the Hamiltonian to satisfy the conditions. The upper bound of the energy error in the regularisation method is $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) - E_g$, up to the failure probability.

4 Analysis of the measurement cost

In this work, we rigorously analyse the number of measurements. Let ϵ be the target error in the ground-state energy. We will give a measurement number sufficient (i.e. an upper bound of the measurement number necessary) for achieving the target. For the regularisation method, we already have an upper bound of the error $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) - E_g$. Therefore, we can derive such a measurement number by solving the equation

$$\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) - E_g = \epsilon, \tag{9}$$

in which η is the unknown. With the solution, we can work out the corresponding measurement number M according to Table 2. If matrix entries are measured independently, the total measurement number is $M_{tot} = 4d^2M$; we have the following theorem.

Theorem 1. *Suppose $\epsilon > \epsilon_K$ and $E_g + \epsilon < 0$. The total measurement number*

$$M_{tot} = \frac{\alpha(\kappa)\beta(d)}{16\eta^2} \tag{10}$$

Protocol	η	α	β	Basis
IM (Chebyshev)	$\frac{2d^2}{\sqrt{M\kappa}}$	$\frac{256}{\kappa}$	d^6	All
IM (Hoeffding)	$\sqrt{\frac{2d^2}{M}\ln\frac{8d^2}{\kappa}}$	$128\ln\frac{1}{\kappa}$	d^4	
CM (Real Hankel)	$\sqrt{\frac{2d}{M}\ln\frac{4d}{\kappa}}$	$64\ln\frac{1}{\kappa}$	$d(2d-1)$	P, GP, IP, ITE
CM (Real symmetric)		$32\ln\frac{1}{\kappa}$	$d^2(d+1)$	CP, F
CM (Complex Toeplitz)	$\sqrt{\frac{2(2d-1)}{M}\ln\frac{4d}{\kappa}}$	$64\ln\frac{1}{\kappa}$	$(2d-1)^2$	RTE

Table 2: The regularisation parameter η, and measurement cost factors $\alpha(\kappa)$ and $\beta(d)$. These quantities depend on the protocol for measuring matrix entries. Two types of protocols are considered: independent measurement (IM) and collective measurement (CM). Using the Chebyshev and Hoeffding inequalities, we obtain two different results (upper bounds) for the IM protocol. These factors are based on the protocols introduced in Appendices B and E. We remark that for CP, leveraging the characteristic of Chebyshev polynomials, the scaling with d can be reduced to $O(d^2)$ [20].

is sufficient for achieving the permissible error ϵ and failure probability κ, i.e. $\hat{E}_{min} \in [E_g, E_g+\epsilon]$ with a probability of at least $1-\kappa$. Here, $\alpha(\kappa) = 256/\kappa$, $\beta(d) = d^6$, and η is the solution to Eq. (9).

See Appendix C for the proof. Similar to Lemma 1, we can always satisfy the condition $E_g+\epsilon<0$ by subtracting a sufficiently large positive constant from the Hamiltonian. The above theorem also holds (up to some approximations) for collective measurements if we replace functions $\alpha(\kappa)$ and $\beta(d)$ according to Table 2.

4.1 Factoring the measurement cost

To compare different bases, we would like to divide the measurement number into three factors. If matrix entries are measured independently, the first two factors are the same for different bases of Krylov subspaces, and the third factor depends on the basis. We find that the third factor is related to the ill-conditioned problem, and we can drastically reduce it with our measurement-efficient algorithm.

We can always write the total measurement number Eq. (10) as a product of three factors,

$$M_{tot} = \frac{\alpha(\kappa)\|H\|_2^2}{p_g^2\epsilon^2} \times \beta(d) \times \gamma. \tag{11}$$

The first factor is the cost of measuring the energy. Roughly speaking, it is the cost in the ideal case that one can prepare the ground state by an ideal projection; see Appendix D. The spectral norm $\|H\|_2$ characterises the range of the energy. Assume that the statistical error and ϵ are comparable, $M_{tot} \propto \|H\|_2^2/\epsilon^2$ according to the standard scaling of the statistical error. $\alpha(\kappa)$ is the overhead for achieving the permissible failure probability κ. The success of KSD algorithms depends on a finite overlap between the reference state and the true ground state $|\psi_g\rangle$ [6, 16, 20]. This requirement is reflected in $M_{tot} \propto 1/p_g^2$, where $p_g = |\langle\psi_g|\varphi\rangle|^2$. The second factor $\beta(d)$ is the overhead due to measuring $d\times d$ matrices. There are more sources of statistical errors (matrix entries) influencing the eventual result when d is larger. The remaining factor

$$\gamma = \frac{p_g^2\epsilon^2}{16\|H\|_2^2\eta^2} \tag{12}$$

is related to the spectrum of $\mathbf{S}$ (i.e. the basis), as we will show in Section 7. Notice that this expression of γ holds for all the bases. In the numerical result, we find that the typical value of γ for certain bases can be as large as 10^{12} to achieve certain ϵ. We propose the Gaussian-power (GP) basis (see Table 1), and it can reduce γ to about 4.

5 Measurement-efficient algorithm

As we have shown, the measurement overhead γ depends on the overlap matrix $\mathbf{S}$, i.e. how we choose the basis of Krylov subspace. The main advantage of our algorithm is that we utilise a basis resulting in a small γ.

To generate a standard Krylov subspace, we

choose operators

$$f_k = (H - E_0)^{k-1} e^{-\frac{1}{2}(H-E_0)^2\tau^2}, \tag{13}$$

where E_0 is a constant up to choice. We call it the GP basis. The corresponding subspace is a conventional Hamiltonian-power Krylov subspace, but the reference state $|\varphi\rangle$ has been *effectively* replaced by $e^{-\frac{1}{2}(H-E_0)^2\tau^2}|\varphi\rangle$.

The spectral norm of f_k operators for the GP basis [defined in Eq. (13)] has the upper bound

$$\|f_k\|_2 \leqslant (\frac{k-1}{e\tau^2})^{\frac{k-1}{2}}, \tag{14}$$

where e is Euler's constant. This upper bound is proven in Lemma 9 in Appendix F. Under the condition $e\tau^2 > d-1$, the norm upper bound $(\frac{k-1}{e\tau^2})^{\frac{k-1}{2}}$ decreasing exponentially with k. This property is essential for the measurement efficiency of our algorithm.

We propose to realise the Gaussian-power basis by expressing the operator of interest as a linear combination of unitaries (LCU). The same approach has been taken to realise other bases like power [19, 12], inverse power [18], imaginary-time evolution [34], filter [15] and Gaussian function of the Hamiltonian [35].

Suppose we want to measure the quantity $\langle\varphi|A|\varphi\rangle$. Here $A = f_k^\dagger H f_q$ and $A = f_k^\dagger f_q$ for $\mathbf{H}_{k,q}$ and $\mathbf{S}_{k,q}$, respectively. First, we work out an expression in the from $A = \sum_s q_s U_s$, where U_s are unitary operators, and q_s are complex coefficients. Then, there are two ways to measure $\langle\varphi|A|\varphi\rangle$. When the expression has finite terms, we can apply A on the state $|\varphi\rangle$ by using a set of ancilla qubits; In the literature, LCU usually refers to this approach [36]. Alternatively, we can evaluate the summation $\langle\varphi|A|\varphi\rangle$ using the Monte Carlo method [37] by averaging Hadamard-test [33] estimates of randomly selected terms $\langle\varphi|U_s|\varphi\rangle$. The second way works for an expression with even infinite terms and requires only one or zero ancilla qubits [38, 39]. We focus on the Monte Carlo method in this work.

Now we give our LCU expression. Suppose we already express H as a linear combination of Pauli operators, it is straightforward to work out the LCU expression of A given the expression of f_k. Therefore, we only give the expression of f_k. Utilising the Fourier transformation of a modified Hermite function (which is proved in Appendix F), we can express f_k in Eq. (13) as

$$f_k = \frac{i^{k-1}}{2^{\frac{k-1}{2}}\tau^{k-1}} \int_{-\infty}^{+\infty} dt H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right) g_\tau(t) \times e^{-ixt}, \tag{15}$$

where $H_n(u)$ denotes Hermite polynomials [40], $g_\tau(t) = \frac{1}{\tau\sqrt{2\pi}} e^{-\frac{t^2}{2\tau^2}}$ is the normalised Gaussian function, $x = H - E_0$, and e^{-ixt} is the real-time evolution (RTE) operator. Notice that in Eq. (15) f_k is expressed as a linear combination of RTE operators e^{-iHt}, and the summation in conventional LCU is replaced with an integral over t. See Appendix G for how to evaluate this integral using the Monte Carlo method.

5.1 Real-time evolution

In order to implement the LCU expression Eq. (15), we need to implement the RTE $e^{-iHt}|\varphi\rangle$. There are various protocols for implementing RTE, including Trotterisation [41–43], LCU [36, 44, 45, 37] and qDRIFT [46], etc. In most of the protocols, RTE is inexact but approaches the exact one when the circuit depth increases. Therefore, all these protocols work for our algorithm as long as the circuit is sufficiently deep. In this work, we specifically consider an LCU-type protocol, the zeroth-order leading-order-rotation formula [47].

In the leading-order-rotation protocol, RTE is *exact* even when the circuit depth is finite. In this protocol, we express RTE in the LCU form

$$e^{-iHt} = \left[\sum_r v_r(t/N) V_r(t/N)\right]^N, \tag{16}$$

where $V_r(t) = e^{-i\phi(t)\sigma}, \sigma$ is a rotation or Pauli operator, $v_r(t)$ are complex coefficients, and N is the time step number. This exact expression of RTE is worked out in the spirit of Taylor expansion and contains infinite terms. Notice that N determines the circuit depth. For details of the zeroth-order leading-order-rotation formula, see Ref. [47] or Appendix G in this paper, where Eq. (77) gives the specific form of Eq. (16). Substituting Eq. (16) into Eq. (15), we obtain the eventual LCU expression of f_k, a concatenation

of two LCU expressions, i.e.

$$f_k = \frac{i^{k-1}}{2^{\frac{k-1}{2}}\tau^{k-1}} \int_{-\infty}^{+\infty} dt H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right) g_\tau(t) e^{iE_0 t} \times \left[\sum_r v_r(t/N) V_r(t/N)\right]^N. \quad (17)$$

With the above expression, we have written f_k as the linear combination of $V_r(t/N)$. Therefore, we can use the Monte Carlo method to realise f_k: We sample t and r according to the coefficients, then we implement the corresponding $V_r(t/N)$ on the quantum computer.

5.2 Bias and variance

Let $\hat{A}$ be the estimator of $\langle\varphi|A|\varphi\rangle$. There are two types of errors in $\hat{A}$, bias and variance. Because RTE is exact in the leading-order-rotation protocol, the estimator $\hat{A}$ is unbiased. Therefore, we can focus on the variance from now on.

If we use the Monte Carlo method and the one-ancilla Hadamard test to evaluate the LCU expression of A, the variance has the upper bound

$$\mathrm{Var}(\hat{A}) \leqslant \frac{2C_A^2}{M}, \quad (18)$$

where $2M$ is the measurement number, and $C_A = \sum_s |q_s|$ is the 1-norm of coefficients in $A = \sum_s q_s U_s$. Here, A is a summation of unitary operators. The generalisation to integral is straightforward. We call C_A the cost of the LCU expression. We remark that the variance in this form is universal and valid for all algorithms with some proper factor C_A.

The cost of an LCU expression is related to the spectral norm. Notice $\|U_s\|_2 = 1$, we immediately conclude that $C_A \geqslant \|A\|_2$. Therefore, the best we can achieve is a carefully designed LCU expression that C_A is as close as possible to $\|A\|_2$. Only when $\|f_k\|_2$ decreases exponentially with k, it is possible to measure $\langle\varphi|A|\varphi\rangle$ with an error decreasing exponentially with k.

In our algorithm, we find that the cost has the form

$$C_A = h_{tot} c_k c_q \text{ and } c_k c_q \quad (19)$$

for $\mathbf{H}_{k,q}$ and $\mathbf{S}_{k,q}$, respectively. c_k is the cost due to f_k, and

$$c_k = \frac{1}{2^{\frac{k-1}{2}}\tau^{k-1}} \int_{-\infty}^{+\infty} dt \left|H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right)\right| g_\tau(t) \times \left[\sum_r \left|v_r\left(\frac{t}{N}\right)\right|\right]^N \leqslant 2\left(\frac{k-1}{e\tau^2}\right)^{\frac{k-1}{2}} \quad (20)$$

This upper bound holds when $N \geqslant 4eh_{tot}^2\tau^2$, which is a requirement on the circuit depth N. The proof is in Appendix G. With the upper bound, we can find that we already approach the lower bound of the variance given by the spectral norm (The difference is a factor of 2). As a result, the variance of $\hat{A}$ decreases exponentially with k and q.

Detailed pseudocodes and analysis are given in Appendix G. There are two ways to apply Theorem 1: First, we can take $C_{\mathbf{H}} = h_{tot} C_{\mathbf{S}}$ and $C_{\mathbf{S}} = \max\{c_k c_q\}$; Second, we can rescale the basis by taking $f'_k = f_k/c_k$. Matrix entries $\hat{\mathbf{H}}'_{k,q}$ and $\hat{\mathbf{S}}'_{k,q}$ of the rescaled basis $\{f'_k|\varphi\rangle\}$ have variance upper bounds $2h_{tot}^2/M$ and $2/M$, respectively. Therefore, $C_{\mathbf{H}} = h_{tot}$ and $C_{\mathbf{S}} = 1$ for the rescaled basis. We take the rescaled basis in the numerical study.

6 Measurement overhead benchmarking

With Theorem 1, we can benchmark the measurement cost in quantum KSD algorithms listed in Table 1. We focus on the overhead factor γ, while ignoring the slight differences in the other two factors in different algorithms (See Table 2). Given a target error ϵ, first we work out the solution η to Eq. (9), then γ is calculated according to Eq. (12).

Two models of strongly correlated systems, namely, the anti-ferromagnetic Heisenberg model and Hubbard model [7, 48–50], are used in benchmarking. For each model, the lattice is taken from two categories: regular lattices, including chain and ladder, and randomly generated graphs. We fix the lattice size such that the Hamiltonian can be encoded into ten qubits. For each Hamiltonian specified by the model and lattice, we evaluate all quantum KSD algorithms with the same subspace dimension d taken from $2, 3, \ldots, 30$. In total, 233 instances of $(model, lattice, d)$ are generated to benchmark

quantum KSD algorithms. The details of the numerical calculations including parameters taken in different algorithms are given in Appendix H.

Fig. 1 illustrates the result of the instance $(\text{Heisenberg}, \text{chain}, d = 5)$, for example. We can find that the error decreases with the cost overhead. When the target error is relatively high, algorithms have similar performance, e.g. when $\epsilon \approx 0.01$, the cost $\gamma \approx 10$ is sufficient in most algorithms. However, in comparison with others, the error in the GP algorithm decreases much faster. When d is sufficiently large, M required to achieve the error ϵ always follows a scaling of $1/\epsilon^2$. However, the size of d at which different algorithms converge to this behavior varies; see Fig. 9. The GP algorithm converges with a smaller d compared to other algorithms. In this instance, the GP algorithm has already converged when $d = 5$, whereas the other algorithms have not yet converged, resulting in poorer performance.

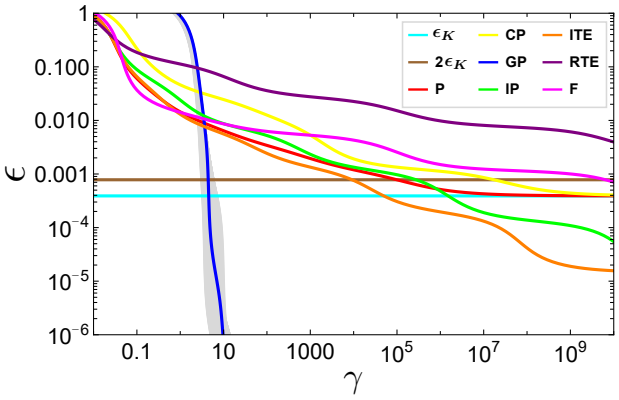


Figure 1: Comparison between quantum KSD algorithms listed in Table 1. Here, we take the instance $(\text{Heisenberg}, \text{chain}, d = 5)$ as an example. ϵ_K is the subspace error of the P algorithm. The grey area illustrates the range of γ when we take $E_0 \in [E_g - 0.1, E_g + 0.1]$ in the GP algorithm. Notice that $\|H\|_2 = 1$. The blue curve represents the result of the GP algorithm with $E_0 = E_g$. Similar results are obtained when other measures are used for benchmarking, including the measurement number M computed with the optimal α and β factors, the necessary measurement cost (instead of its upper bound) estimated numerically and the necessary cost when using the thresholding method; see Appendix L.

To confirm this advantage of the GP algorithm, we implement the simulation for a number of instances. We choose the power algorithm as the standard for comparison. We remark that the power algorithm and the classical Lanczos algorithm yield the same result when the implementation is error-free (i.e. without statistical error and rounding error), and the eventual error in this ideal case is the subspace error ϵ_K. Given the model, lattice and subspace dimension, we compute ϵ_K of the power algorithm. Then, we take the permissible error $\epsilon = 2\epsilon_K$ and compute the overhead factor γ for each algorithm. Then, the empirical distribution of γ is shown in Fig. 2.

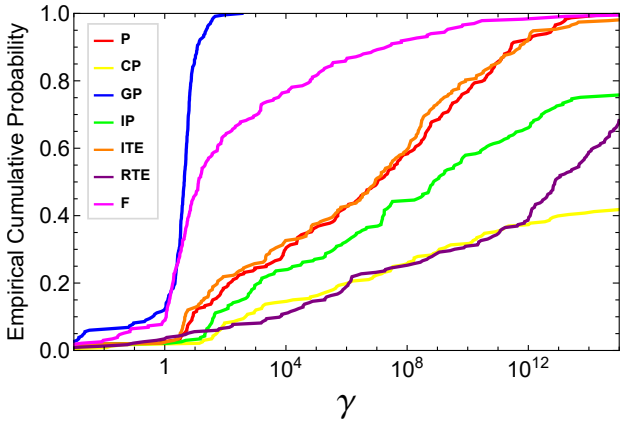


Figure 2: Empirical distribution of the measurement overhead γ for algorithms listed in Table 1. In the Gaussian-power algorithm, we take a random E_0 in the interval $[E_g - 0.1\|H\|_2, E_g + 0.1\|H\|_2]$, i.e. we assume that we have a preliminary estimation of the ground-state energy with an uncertainty as large as 10% of the entire spectrum.

From the distribution, we can find that our Gaussian-power algorithm has the smallest measurement overhead, and γ is smaller than 10^2 for almost all instances. The filter (F) algorithm is the most well-performed one among others. Part of the reason is that we have taken the optimal parameter found by grid search, of which the details are discussed in Appendix H. The median value of γ is about 4 for the GP algorithm, 13 for the F algorithm and as large as 10^{12} for some other algorithms. Comparing the GP and F algorithms, although the median values are similar, the cost overhead is larger than 10^4 in more than 20% instances in the F algorithm, which never happens in the GP algorithm.

The performance of the GP algorithm depends on the choice of parameters: τ, E_0 and η. The regularisation parameter η is taken according to Table 2. To choose the value of E_0, we need prior knowledge about the true ground-state energy E_g, which can be obtained from certain classical algorithms computing the ground state or the

quantum KSD algorithm taking a smaller subspace dimension. Although we prefer an E_0 close to E_g, the numerical result suggests that a relatively large difference $|E_0 - E_g|$ (10% of the entire spectrum) is tolerable. The Gaussian factor in the GP basis acts as a filter on the spectrum. Therefore, if $|E_0 - E_g|$ is finite and we take an over large τ, we may have a vanishing overlap between the effective reference state $e^{-\frac{1}{2}(H-E_0)^2\tau^2}|\varphi\rangle$ and the ground state. To avoid this problem, we need to assume an upper bound $|E_0 - E_g| \leqslant \epsilon_0$, then we suggest taking τ in the range $\sqrt{\frac{d-1}{e}} < \tau \leqslant \frac{\sqrt{d-1}}{\epsilon_0}$, see Appendix K. Although we take the same τ for all f_k in each instance in the benchmarking, one can flexibly choose different τ in practice.

7 Composing a projector

We show that γ is related to the spectrum of $\mathbf{S}$. Let's consider a small η and the Taylor expansion of E'. The minimum value of the zeroth-order term $\frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}$ is E_{min} according to the Rayleigh quotient theorem [51]. Take this minimum value, we have $E' \simeq E_{min} + s\eta$, where $s \propto \frac{\mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}$ is the first derivative. The solution to $E' = E_g + \epsilon$ is $\eta \simeq (E_g + \epsilon - E_{min})/s$, then

$$\gamma \simeq u^2 \left(\frac{p_g \mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}\right)^2, \tag{21}$$

where $u \leqslant \frac{(C_\mathbf{H} + \|H\|_2 C_\mathbf{S})\epsilon}{2\|H\|_2(\epsilon - \epsilon_K)} \approx C_\mathbf{S}$ under assumptions $C_\mathbf{H} \approx \|H\|_2 C_\mathbf{S}$ and $\epsilon - \epsilon_K \approx \epsilon$. Eq. (21) is derived in Appendix I. Therefore, when $\mathbf{S}$ is close to singular, the overhead can be large.

We can explain the measurement efficiency of our algorithm by composing a projector. In our algorithm with the rescaled basis, the solution is a state in the form $|\Psi(\mathbf{a})\rangle = \sum_{k=1}^d a_k c_k^{-1} f_k |\varphi\rangle$. Ideally, the linear combination of f_k realises a projector onto the ground state, i.e. $\sum_{k=1}^d a_k c_k^{-1} f_k = |\psi_g\rangle\langle\psi_g|$. Then, $|\Psi(\mathbf{a})\rangle = \sqrt{p_g}|\psi_g\rangle$ up to a phase and $\mathbf{a}^\dagger\mathbf{S}\mathbf{a} = \langle\Psi(\mathbf{a})|\Psi(\mathbf{a})\rangle = p_g$. Then $\gamma \approx \left(\frac{p_g \mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}\right)^2 = \left(\mathbf{a}^\dagger\mathbf{a}\right)^2$ (notice that $C_\mathbf{S} = 1$), we can find that $\mathbf{a}^\dagger\mathbf{a}$ determines γ. When c_k is smaller, $|a_k|$ required for composing the projector is smaller. In our algorithm, c_k decreases exponentially with k, and the speed of decreasing is controllable via the parameter τ. In this way, our algorithm results in a small γ.

To be concrete, let's consider a classic way of composing a projector from Hamiltonian powers using Chebyshev polynomials (CPs) T_n, which has been used for proving the convergence of the Lanczos algorithm [52], RTE and CP quantum KSD algorithms [16, 20]. Without loss of generality, we suppose $\|H\|_2 = 1$. An approximate projector in the form of Chebyshev polynomials is

$$\frac{T_n(Z)}{T_n(z_1)} = |\psi_g\rangle\langle\psi_g| + \Omega, \tag{22}$$

where $Z = 1 - (H - E_g - \Delta)$, Δ is the energy gap between the ground state and the first excited state, $z_1 = 1 + \Delta$, and Ω is an operator with the upper bound $\|\Omega\|_2 \leqslant 2/(z_1^n + z_1^{-n})$. Notice that the error Ω depends on n, and its upper bound decreases exponentially with n if Δ is finite. For simplicity, we focus on the case that $E_0 = E_g$ in the Hamiltonian power. Then, in the expansion $T_n(Z) = \sum_{l=0}^n b_l (H - E_g)^l$, coefficients have upper bounds $|b_l| \leqslant n^l T_n(z_1)$ increasing exponentially with l. See Appendix J. In the LCU and Monte Carlo approach, large coefficients lead to large variance. Therefore, it is difficult to realise this projector because of the exponentially increasing coefficients.

In our algorithm, the exponentially decreasing c_k can cancel out the large b_l. We can compose the CP projector with an additional Gaussian factor. Taking $d = n+1$ and $a_k = c_k b_{k-1}/T_n(z_1)$, we have $\sum_{k=1}^d a_k c_k^{-1} f_k = \frac{T_n(Z)}{T_n(z_1)} e^{-\frac{1}{2}(H-E_g)^2\tau^2}$. Because the Gaussian factor is also a projector onto the ground state, the overall operator is a projector with a smaller error than the CP projector. The corresponding overhead factor is

$$\gamma \lesssim 4 \frac{1}{1 - n^3/(e\tau^2)}. \tag{23}$$

When τ is sufficiently large, $\gamma \lesssim 4$. This example shows that the Gaussian-power basis can compose a projector with a small measurement cost.

By composing a projector with an explicit expression, we can obtain the worst-case performance of the algorithm. Similar analyses are reported in Refs. [16] and [20] for RTE and CP algorithms, respectively. Because the projector is approximate, it results in a finite error in the ground-state energy ϵ_P. By solving the generalised eigenvalue problem, we can usually work out a better solution, i.e. the corresponding energy error is lower than ϵ_P. Our result about the

measurement cost is applicable to a target error that is not limited by ϵ_P. Theorem 1 holds even when $\epsilon < \epsilon_P$, which is beyond the explicit projector analysis.

The method of constructing polynomial projection operators mentioned above can be applied not only to the GP algorithm but also to the P algorithm. Since the variance of the P algorithm does not decrease exponentially with k, it can be seen that our algorithm has an advantage over the P algorithm in this regard. We remark that for this particular projector, the decomposition into Hamiltonian powers is costly, and the existence of a better decomposition is possible. Compared to other algorithms, we have provided a numerical benchmark test in the previous section. Additionally, for the CP and RTE algorithms, it is possible to construct approximate projection operators with constant bounded expansion coefficients [16, 20]. In this case, further methods are needed to theoretically explain the improvement in measurement cost of our algorithm compared to these algorithms.

8 Discussions and Conclusions

We can apply Theorem 1 to estimate the measurement cost of various quantum KSD algorithms for arbitrary target error ϵ and subspace dimension d. In this paper, we focus on a target error comparable to the Lanczos algorithm at the same d, i.e. $\epsilon \sim \epsilon_K$. The Lanczos algorithm is a well-known classical algorithm. It is natural to demand quantum KSD algorithms, as quantum counterparts of Lanczos algorithm, to reach a comparable target error. In this case, the measurement cost of the GP algorithm is observed much smaller than the others in the benchmarking.

In certain cases of practical applications, there is definite target error epsilon, e.g. the chemical accuracy. In these cases, these target errors are no longer depending on ϵ_K. According to the Kaniel–Paige convergence theory [52], by increasing the subspace dimension d and measurement number M, we can always achieve arbitrary finite target error. Our results focus on relatively small d, in which case the GP algorithm requires a much smaller M than the others, as illustrated in Fig. 2. We show in Fig. 9 that M converges to $O(1/\epsilon^2)$ when d is sufficiently large, as demonstrated in Refs. [20, 53]. Increasing d can weaken the advantage of the GP algorithm in the measurement cost. However, increasing d not only brings more measurement entries (i.e. a larger factor β), but also implies deeper quantum circuits and more gate noises.

In conclusion, we have proposed a regularised estimator of the ground-state energy (see Algorithm 1). Even in the presence of statistical errors, the regularised estimator is variational (i.e. equal to or above the true ground-state energy) and has a rigorous upper bound. Because of these properties, it provides a universal way of analysing the measurement cost in quantum KSD algorithms, with the result summarised in Theorem 1. This approach can be generalised to analyse the effect of other errors, e.g. imperfect quantum gates. The robustness to statistical errors in our algorithm implies the robustness to other errors.

To minimise the measurement cost, we have proposed a protocol for constructing Hamiltonian power with an additional Gaussian factor. This Gaussian factor is important because it leads to the statistical error decreasing exponentially with the power. Consequently, we can construct a Chebyshev-type projector onto the ground state with a small measurement cost. We benchmark quantum KSD algorithms with two models of quantum many-body systems. We find that our algorithm requires the smallest measurement number, and a measurement overhead γ of a hundred is almost always sufficient for approaching the theoretical-limit accuracy ϵ_K of the Lanczos algorithm. In addition to the advantage over the classical Lanczos algorithm in scalability, our result suggests that the quantum algorithm is also competitive in accuracy at a reasonable measurement cost.

This paper primarily focuses on the ground-state problem. Notably, our quantum KSD algorithm holds promise for application to low-lying excited states as well. We leave it for future research.

Acknowledgments

YL thanks Xiaoting Wang and Zhenyu Cai for the helpful discussions. This work is supported by the National Natural Science Foundation of China (Grants No. 12225507 and No. 12088101), Inno-

vation Program for Quantum Science and Technology (Grant No. 2023ZD0300200), and NSAF (Grants No. U2330201 and No. U2330401). The source codes for the numerical simulation are available at GitHub [54].

A General formalism of KSD algorithm

Given a Hamiltonian H, a reference state $|\varphi\rangle$ and an integer d, the standard Krylov subspace is $\mathcal{K} = \text{Span}\left(|\varphi\rangle, H|\varphi\rangle, H^2|\varphi\rangle, \ldots, H^{d-1}|\varphi\rangle\right)$. This can be seen as taking polynomials $f_k(x) = x^{k-1}$. The subspace is the same when $\{f_1, f_2, \ldots, f_d\}$ is an arbitrary set of linearly-independent $(d-1)$-degree polynomials. Any state in the Krylov subspace can be expressed as

$$|\psi_K\rangle \quad = \quad \sum_{k=1}^{d} a_k f_k(H)|\varphi\rangle. \tag{24}$$

The state that minimises the Rayleigh quotient $\langle\psi_K|H|\psi_K\rangle/\langle\psi_K|\psi_K\rangle$ is regarded as the approximate ground state of the Hamiltonian H, and the corresponding energy is the approximate ground-state energy, i.e.

$$|\psi_g\rangle \quad \sim \quad |\psi_{min}\rangle = \arg\min_{\psi_K\in\mathcal{K}} \frac{\langle\psi_K|H|\psi_K\rangle}{\langle\psi_K|\psi_K\rangle}, \tag{25}$$

$$E_g \quad \sim \quad E_{min} = \min_{\psi_K\in\mathcal{K}} \frac{\langle\psi_K|H|\psi_K\rangle}{\langle\psi_K|\psi_K\rangle}. \tag{26}$$

Definition 1 (Rayleigh quotient)**.** For any matrix $\mathbf{A} \in \mathbb{C}^{d\times d}$, and any non-zero vector $\mathbf{a} \in \mathbb{C}^d$, the Rayleigh quotient is defined as

$$\langle\mathbf{A}\rangle(\mathbf{a}) \quad = \quad \frac{\mathbf{a}^\dagger\mathbf{A}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{a}}. \tag{27}$$

The approximate ground state energy can be rewritten as

$$E_{min} \quad = \quad \min_{\mathbf{a}\neq 0} \frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}} = \min_{\mathbf{a}\neq 0} \frac{\langle\mathbf{H}\rangle(\mathbf{a})}{\langle\mathbf{S}\rangle(\mathbf{a})}, \tag{28}$$

where $\mathbf{H}$ and $\mathbf{S}$ are $d \times d$ Hermitian matrices as defined in the main text. Notice that E_{min} is the minimum eigenvalue of Eq. (2).

When k increases, $H^k|\varphi\rangle$ converges to the eigenstate (having non-zero overlap with $|\varphi\rangle$) with the largest absolute eigenvalue. The eigenstate is usually the ground state (If not, take $H \leftarrow H - E_0$ where E_0 is a positive constant). Therefore, it is justified to express the ground state as a linear combination of $\{H^k|\varphi\rangle\}$ as long as $|\varphi\rangle$ has a non-zero overlap with the true ground state $|\psi_g\rangle$. However, the convergence with k causes inherent trouble that basis vectors of the Krylov subspace are nearly linearly dependent. Then, the overlap matrix $\mathbf{S}$, which is a Gram matrix, becomes nearly singular and has a large condition number. As a result, the denominator of the Rayleigh quotient can be very small, and a tiny error in computing may cause a large deviation of the approximate ground-state energy E_{min}.

According to the Rayleigh-Ritz theorem (Rayleigh quotient theorem) [51], the minimisation problem is equivalent to the generalised eigenvalue problem Eq. (2). The generalised eigenvalue problem can be reduced to an eigenvalue problem while the singularity issue remains. For example, one can use the Cholesky decomposition to express $\mathbf{S}$ as the product of an upper-triangular matrix and its conjugate transpose, i.e. $\mathbf{S} = \mathbf{R}^\dagger\mathbf{R}$, and the eigenvalue problem becomes $\left(\mathbf{R}^\dagger\right)^{-1}\mathbf{H}\mathbf{R}^{-1}(\mathbf{R}\mathbf{a}) = E\mathbf{R}\mathbf{a}$. The Cholesky decomposition, however, also requires the matrix $\mathbf{S}$ to be positive-definite. When $\mathbf{S}$ is nearly singular, the Cholesky decomposition is unstable. The QZ algorithm is a more numerically stable method, and it is built into most standard programming languages. One way to address the singularity issue is by adding a positive diagonal matrix to $\mathbf{S}$. In this work, we also add a diagonal matrix to $\mathbf{H}$ to ensure that the Rayleigh quotient is variational, i.e. the energy is not lower than the ground-state energy (with a controllable probability). An alternative way to address the singularity issue is the so-called thresholding method, see Refs. [8, 16].

B Norm distributions of noise matrices

In this section, we analyse the distributions of $\|\hat{\mathbf{H}} - \mathbf{H}\|_2$ and $\|\hat{\mathbf{S}} - \mathbf{S}\|_2$. For the two matrices, there are a total of $2d^2$ matrix entries. We can measure each of them independently on the quantum computer. Under this independent measurement protocol, we obtain the most general results that apply to all the bases in Table 1, which are summarised in Lemma 2 and Lemma 3. In the $2d^2$ matrix entries, some of them are equivalent. To reduce the measurement cost, we can only measure one entry in each set of equivalent

matrix entries, and we call it the collective measurement protocol. Because the equivalent-entry sets depend on the basis, the corresponding distributions are basis-dependent. The results are summarised in Lemma 4 and Lemma 5; we remark that these results are obtained under the assumption that statistical noise is Gaussian.

Lemma 2 (Based on Chebyshev's inequality)**.** *Let κ be the failure probability. The inequality (6) holds when $\eta\sqrt{M} \geqslant \frac{2d^2}{\sqrt{\kappa}}$.*

Proof. Recall the variance upper bounds of $\hat{\mathbf{H}}_{k,q}$ and $\hat{\mathbf{S}}_{k,q}$ given in Eq. (4) and (5), respectively. When $M \geqslant \frac{4d^4}{\kappa\eta^2}$,

$$\mathrm{Var}(\hat{\mathbf{H}}_{k,q}) \leqslant \frac{\kappa C_{\mathbf{H}}^2 \eta^2}{2d^4}, \tag{29}$$

$$\mathrm{Var}(\hat{\mathbf{S}}_{k,q}) \leqslant \frac{\kappa C_{\mathbf{S}}^2 \eta^2}{2d^4}. \tag{30}$$

According to Chebyshev's inequality, we have

$$\Pr\left(|\hat{\mathbf{H}}_{k,q} - \mathbf{H}_{k,q}| \geqslant \frac{C_{\mathbf{H}}\eta}{d}\right) \leqslant \frac{\kappa}{2d^2}, \tag{31}$$

$$\Pr\left(|\hat{\mathbf{S}}_{k,q} - \mathbf{S}_{k,q}| \geqslant \frac{C_{\mathbf{S}}\eta}{d}\right) \leqslant \frac{\kappa}{2d^2}. \tag{32}$$

Since matrix entries are measured independently, estimators $\hat{\mathbf{H}}_{k,q}$ or $\hat{\mathbf{S}}_{k,q}$ are independent. We have

$$\begin{aligned} &\Pr\left(\begin{array}{c} \forall k,q\ |\hat{\mathbf{H}}_{k,q} - \mathbf{H}_{k,q}| \leqslant \frac{C_{\mathbf{H}}\eta}{d} \\ \text{and } |\hat{\mathbf{S}}_{k,q} - \mathbf{S}_{k,q}| \leqslant \frac{C_{\mathbf{S}}\eta}{d} \end{array}\right) \\ \geqslant &\left[\left(1 - \frac{\kappa}{2d^2}\right)^{d^2}\right]^2 \geqslant 1 - \kappa, \end{aligned} \tag{33}$$

where Bernoulli's inequality is used.

Let $\mathbf{A}$ be an $m \times n$ matrix with entries $A_{i,j}$, its spectral norm satisfies

$$\|A\|_2 \leqslant \sqrt{mn}\max_{i,j}|A_{i,j}|. \tag{34}$$

This is because of the well known relation $\|A\|_2 \leqslant \|A\|_F$, where the Frobenius norm is $\|A\|_F = \left(\sum_{i=1}^m \sum_{j=1}^n |A_{i,j}|^2\right)^{1/2}$. Therefore, when $|\hat{\mathbf{H}}_{k,q} - \mathbf{H}_{k,q}| \leqslant \frac{C_{\mathbf{H}}\eta}{d}$ and $|\hat{\mathbf{S}}_{k,q} - \mathbf{S}_{k,q}| \leqslant \frac{C_{\mathbf{S}}\eta}{d}$ for all k and q, we have $\|\hat{\mathbf{H}} - \mathbf{H}\|_2 \leqslant C_{\mathbf{H}}\eta$ and $\|\hat{\mathbf{S}} - \mathbf{S}\|_2 \leqslant C_{\mathbf{S}}\eta$. □

The above proof applies to all bases and measurement protocols and holds without any assumption on the statistical noise. As the most general result, Lemma 2 is used to prove Theorem 1 in Appendix C.

To measure matrix entries on a quantum computer, a practical method is using Hadamard-test circuits. Then, measurement outcomes are independent random variables ± 1, i.e. the random variables are bounded even taking into account other factors (We will give an example later, see Algorithm 2, where the phase of q_s can be absorbed into U_s). In this case, we can use Hoeffding's inequality [55] to obtain an improved result.

Lemma 3 (Based on Hoeffding's inequality)**.** *Suppose matrix entries $\hat{\mathbf{H}}_{k,q}$ and $\hat{\mathbf{S}}_{k,q}$ are empirical means of random variables in ranges $[-C_{\mathbf{H}}, C_{\mathbf{H}}]$ and $[-C_{\mathbf{S}}, C_{\mathbf{S}}]$, respectively. Let κ be the failure probability. If the matrix elements of $\mathbf{H}$ and $\mathbf{S}$ are estimated using Algorithm 2, then the inequality (6) holds when $\eta\sqrt{M} \geqslant \sqrt{2d^2 \ln\frac{8d^2}{\kappa}}$.*

Proof. For each matrix entry, its real (imaginary) part is computed by taking the mean of M random variables. According to Hoeffding's inequality, we have,

$$\begin{aligned} &\Pr\left(|\hat{\mathbf{H}}_{k,q} - \mathbf{H}_{k,q}| \geqslant \frac{C_{\mathbf{H}}\eta}{d}\right) \\ \leqslant &\Pr\left(|\mathrm{Re}(\hat{\mathbf{H}}_{k,q}) - \mathrm{Re}(\mathbf{H}_{k,q})| \geqslant \frac{C_{\mathbf{H}}\eta}{\sqrt{2}d}\right) \\ \times &\Pr\left(|\mathrm{Im}(\hat{\mathbf{H}}_{k,q}) - \mathrm{Im}(\mathbf{H}_{k,q})| \geqslant \frac{C_{\mathbf{H}}\eta}{\sqrt{2}d}\right) \\ \leqslant &\ 4e^{-\frac{M\eta^2}{2d^2}}. \end{aligned} \tag{35}$$

Similarly,

$$\Pr\left(|\hat{\mathbf{S}}_{k,q} - \mathbf{S}_{k,q}| \geqslant \frac{C_{\mathbf{S}}\eta}{d}\right) \leqslant 4e^{-\frac{M\eta^2}{2d^2}}. \tag{36}$$

When $M \geqslant \frac{2d^2}{\eta^2}\ln\frac{8d^2}{\kappa}$, Eq. (31) and Eq. (32) hold. The rest of the proof is the same as in Lemma 2. □

Considering the equivalence between matrix entries, we can find that $\mathbf{H}$ and $\mathbf{S}$ are real Hankel matrices for the GP and P, IP and ITE bases in Table 1, real symmetric matrices for the CP and F bases and complex Hermitian-Toeplitz matrices for the RTE basis. Under the collective measurement protocol, random matrices $\hat{\mathbf{H}} - \mathbf{H}$ and $\hat{\mathbf{S}} - \mathbf{S}$ are matrices of the same type as $\mathbf{H}$ and $\mathbf{S}$. To analyse the distribution of these random matrices, we assume that the distributions of

$\hat{\mathbf{H}}_{k,q}-\mathbf{H}_{k,q}$ and $\hat{\mathbf{S}}_{k,q}-\mathbf{S}_{k,q}$ are Gaussian with the zero mean. Notice that inequivalent entries are independent random variables. Then, we obtain the following results.

Lemma 4 (Real symmetric/Hankel matrices)**.** *Let κ be the failure probability. If $\hat{\mathbf{H}}-\mathbf{H}$ and $\hat{\mathbf{S}}-\mathbf{S}$ are random real symmetric matrices following the Gaussian distribution, then the inequality (6) holds when $\eta\sqrt{M}\geqslant\sqrt{2d\ln\frac{4d}{\kappa}}$.*

Proof. Applying the matrix Gaussian series theorem [56] to the real symmetric matrix with Gaussian noise, we have

$$\Pr\left(\|\hat{\mathbf{H}}-\mathbf{H}\|_2\geqslant C_{\mathbf{H}}\eta\right)\leqslant 2de^{-\frac{M\eta^2}{2d}}, \tag{37}$$

where $\mathrm{Var}(\hat{\mathbf{H}}_{k,q})\leqslant\frac{C_{\mathbf{H}}^2}{M}$ is used. Similarly,

$$\Pr\left(\|\hat{\mathbf{S}}-\mathbf{S}\|_2\geqslant C_{\mathbf{S}}\eta\right)\leqslant 2de^{-\frac{M\eta^2}{2d}}. \tag{38}$$

Note that the real Hankel matrices with Gaussian noise yield the same inequalities [57].

To make sure $\|\hat{\mathbf{H}}-\mathbf{H}\|_2\leqslant C_{\mathbf{H}}\eta$ and $\|\hat{\mathbf{S}}-\mathbf{S}\|_2\leqslant C_{\mathbf{S}}\eta$ with a probability of at least $1-\kappa$, we take M such that

$$2de^{-\frac{M\eta^2}{2d}}=\frac{\kappa}{2}. \tag{39}$$

Then,

$$M=\frac{2d}{\eta^2}\ln\frac{4d}{\kappa}. \tag{40}$$

□

Lemma 5 (Complex Hermitian-Toeplitz matrices)**.** *Let κ be the failure probability. If $\hat{\mathbf{H}}-\mathbf{H}$ and $\hat{\mathbf{S}}-\mathbf{S}$ are random complex Hermitian-Toeplitz matrices following the Gaussian distribution, then the inequality (6) holds when $\eta\sqrt{M}\geqslant\sqrt{2(2d-1)\ln\frac{4d}{\kappa}}$.*

Proof. For complex Hermitian-Toeplitz matrices, the matrix Gaussian series theorem [56] gives

$$\Pr\left(\|\hat{\mathbf{H}}-\mathbf{H}\|_2\geqslant C_{\mathbf{H}}\eta\right)\leqslant 2de^{-\frac{M\eta^2}{2(2d-1)}}. \tag{41}$$

Similarly,

$$\Pr\left(\|\hat{\mathbf{S}}-\mathbf{S}\|_2\geqslant C_{\mathbf{S}}\eta\right)\leqslant 2de^{-\frac{M\eta^2}{2(2d-1)}}. \tag{42}$$

To make sure $\|\hat{\mathbf{H}}-\mathbf{H}\|_2\leqslant C_{\mathbf{H}}\eta$ and $\|\hat{\mathbf{S}}-\mathbf{S}\|_2\leqslant C_{\mathbf{S}}\eta$ with a probability of at least $1-\kappa$, we take

$$M=\frac{2(2d-1)}{\eta^2}\ln\frac{4d}{\kappa}. \tag{43}$$

□

C Details on the analysis of measurement cost

In this section, we develop theoretical tools for analysing the measurement cost in quantum KSD algorithms. At the end of this section, we give the proofs of Lemma 1 and Theorem 1.

Definition 2. For simplification, we define the following functions:

$$E(\mathbf{a}) = \frac{\langle\mathbf{H}\rangle(\mathbf{a})}{\langle\mathbf{S}\rangle(\mathbf{a})}, \tag{44}$$

$$\hat{E}(\eta,\mathbf{a}) = \frac{\langle\hat{\mathbf{H}}\rangle(\mathbf{a})+C_{\mathbf{H}}\eta}{\langle\hat{\mathbf{S}}\rangle(\mathbf{a})+C_{\mathbf{S}}\eta}, \tag{45}$$

$$E'(\eta,\mathbf{a}) = \frac{\langle\mathbf{H}\rangle(\mathbf{a})+2C_{\mathbf{H}}\eta}{\langle\mathbf{S}\rangle(\mathbf{a})+2C_{\mathbf{S}}\eta}. \tag{46}$$

Here, functions $\langle\mathbf{A}\rangle(\mathbf{a})$ are defined in Eq. (27). $\min_{\mathbf{a}\neq 0}E(\mathbf{a})=E_{min}$ is the minimum eigenvalue of the generalised eigenvalue problem in Eq. (2); $\min_{\mathbf{a}\neq 0}\hat{E}(\eta,\mathbf{a})=\hat{E}_{min}$ is the minimum eigenvalue of the generalised eigenvalue problem solved in Algorithm 1; and Eq. (46) is equivalent to Eq. (7).

Lemma 6. *If $\|\hat{\mathbf{H}}-\mathbf{H}\|_2\leqslant C_{\mathbf{H}}\eta$ and $\|\hat{\mathbf{S}}-\mathbf{S}\|_2\leqslant C_{\mathbf{S}}\eta$, the following statements hold:*

(1) $\hat{E}(\eta,\mathbf{a})\geqslant\min\{E(\mathbf{a}),0\}$;

(2) If $\hat{E}(\eta,\mathbf{a})<0$, then $\hat{E}(\eta,\mathbf{a})\leqslant E'(\eta,\mathbf{a})$;

(3) If $E'(\eta,\mathbf{a})<0$, then $\hat{E}(\eta,\mathbf{a})<0$.

Proof. Consider the error in the denominator,

$$\begin{aligned}|\langle\hat{\mathbf{S}}\rangle(\mathbf{a})-\langle\mathbf{S}\rangle(\mathbf{a})| &= \left|\frac{\mathbf{a}^\dagger\hat{\mathbf{S}}\mathbf{a}-\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{a}}\right| \\ &= \left|\frac{\mathbf{a}^\dagger(\hat{\mathbf{S}}-\mathbf{S})\mathbf{a}}{\mathbf{a}^\dagger\mathbf{a}}\right|\leqslant\frac{|\mathbf{a}^\dagger|\|\hat{\mathbf{S}}-\mathbf{S}\|_2|\mathbf{a}|}{\mathbf{a}^\dagger\mathbf{a}} \\ &\leqslant C_{\mathbf{S}}\eta,\end{aligned} \tag{47}$$

from which we obtain

$$\begin{aligned}0<\langle\mathbf{S}\rangle(\mathbf{a}) &\leqslant \langle\hat{\mathbf{S}}\rangle(\mathbf{a})+C_{\mathbf{S}}\eta \\ &\leqslant \langle\mathbf{S}\rangle(\mathbf{a})+2C_{\mathbf{S}}\eta.\end{aligned} \tag{48}$$

Similarly, we have

$$\begin{aligned}\langle\mathbf{H}\rangle(\mathbf{a}) &\leqslant \langle\hat{\mathbf{H}}\rangle(\mathbf{a})+C_{\mathbf{H}}\eta \\ &\leqslant \langle\mathbf{H}\rangle(\mathbf{a})+2C_{\mathbf{H}}\eta.\end{aligned} \tag{49}$$

Let a, b, c and d be positive numbers, it can be shown that

$$\frac{-a}{b} < \frac{-a+c}{b+d}. \tag{50}$$

If $\hat{E}(\eta, \mathbf{a}) < 0$, then $\langle \hat{\mathbf{H}} \rangle(\mathbf{a}) + C_{\mathbf{H}}\eta < 0$. Under this condition, by using Eqs. (48)-(50), we get

$$\begin{aligned} \frac{\langle \mathbf{H} \rangle(\mathbf{a})}{\langle \mathbf{S} \rangle(\mathbf{a})} &\leqslant \frac{\langle \hat{\mathbf{H}} \rangle(\mathbf{a}) + C_{\mathbf{H}}\eta}{\langle \hat{\mathbf{S}} \rangle(\mathbf{a}) + C_{\mathbf{S}}\eta} \\ &\leqslant \frac{\langle \mathbf{H} \rangle(\mathbf{a}) + 2C_{\mathbf{H}}\eta}{\langle \mathbf{S} \rangle(\mathbf{a}) + 2C_{\mathbf{S}}\eta}, \end{aligned} \tag{51}$$

i.e. $E(\mathbf{a}) \leqslant \hat{E}(\eta, \mathbf{a}) \leqslant E'(\eta, \mathbf{a})$. Both the first and second statements are proven.

If $E'(\eta, \mathbf{a}) < 0$, then $\langle \hat{\mathbf{H}} \rangle(\mathbf{a}) + C_{\mathbf{H}}\eta \leqslant \langle \mathbf{H} \rangle(\mathbf{a}) + 2C_{\mathbf{H}}\eta < 0$. Therefore, $\hat{E}(\eta, \mathbf{a}) < 0$. The third statement is proven. □

Under the condition $E'(\eta, \mathbf{a}) < 0$, Lemma 6 states $E(\mathbf{a}) \leqslant \hat{E}(\eta, \mathbf{a}) \leqslant E'(\eta, \mathbf{a})$, where the first inequality shows that the algorithm is variational, and the second inequality gives the error bound. In the following, we will show that this relation still holds after minimisation.

Lemma 7. *Under conditions*

(1) $\|\hat{\mathbf{H}} - \mathbf{H}\|_2 \leqslant C_{\mathbf{H}}\eta$ *and* $\|\hat{\mathbf{S}} - \mathbf{S}\|_2 \leqslant C_{\mathbf{S}}\eta$, *and*

(2) $E_g < 0$ *and* $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) < 0$,

the following statement holds,

$$E_g \leqslant \min_{\mathbf{a}\neq\mathbf{0}} \hat{E}(\eta, \mathbf{a}) \leqslant \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}). \tag{52}$$

Proof. According to the first statement in Lemma 6, $\forall \mathbf{a} \neq \mathbf{0}$, $\hat{E}(\eta, \mathbf{a}) \geqslant \min\{E(\mathbf{a}), 0\}$. Since $E(\mathbf{a}) \geqslant E_g$ and $0 > E_g$, we have $\min_{\mathbf{a}\neq\mathbf{0}} \hat{E}(\eta, \mathbf{a}) \geqslant E_g$.

Let $\mathbf{a}^* = \arg\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a})$. According to the third statement in Lemma 6, the condition $E'(\eta, \mathbf{a}^*) < 0$ implies that $\hat{E}(\eta, \mathbf{a}^*) < 0$. Under this condition, according to the second statement in Lemma 6, $\hat{E}(\eta, \mathbf{a}^*) \leqslant E'(\eta, \mathbf{a}^*)$. Therefore, we have

$$\begin{aligned} &\min_{\mathbf{a}\neq\mathbf{0}} \hat{E}(\eta, \mathbf{a}) \leqslant \hat{E}(\eta, \mathbf{a}^*) \\ \leqslant\ & E'(\eta, \mathbf{a}^*) = \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}). \end{aligned} \tag{53}$$

□

Lemma 8. *Suppose* $\mathbf{S}$ *is invertible,* $\epsilon > \epsilon_K$ *and* $E_g + \epsilon < 0$. *Eq. (9) has a positive solution in* η, *and the solution is unique.*

Proof. Let $\mathbf{a}^*_\eta = \arg\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a})$.

First, we prove that $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a})$ is a strictly monotonically increasing function of η when η is in the range $\{\eta \geqslant 0 | \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) < 0\}$. Let η_2 and η_1 be in the range and $\eta_2 > \eta_1$. By Eq. (50), we have $E'(\eta_1, \mathbf{a}^*_{\eta_2}) < E'(\eta_2, \mathbf{a}^*_{\eta_2})$. Since $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta_1, \mathbf{a}) \leqslant E'(\eta_1, \mathbf{a}^*_{\eta_2})$,

$$\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta_1, \mathbf{a}) < \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta_2, \mathbf{a}). \tag{54}$$

Second, we prove that $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a})$ is a continuous function of η when $\eta \geqslant 0$. When $\eta \geqslant 0$ and $\mathbf{a} \neq \mathbf{0}$, $E'(\eta, \mathbf{a})$ is a continuous function of η. Consequently, $\forall \delta_E > 0$, $\exists \delta_\eta > 0$ such that $E'(\eta + \delta_\eta, \mathbf{a}^*_\eta) - E'(\eta, \mathbf{a}^*_\eta) < \delta_E$. Then,

$$\begin{aligned} 0 &< E'(\eta + \delta_\eta, \mathbf{a}^*_{\eta+\delta_\eta}) - E'(\eta, \mathbf{a}^*_\eta) \\ &< E'(\eta + \delta_\eta, \mathbf{a}^*_\eta) - E'(\eta, \mathbf{a}^*_\eta) \leqslant \delta_E. \end{aligned} \tag{55}$$

Therefore, $\forall \delta_E > 0$, $\exists \delta_\eta > 0$ such that

$$\begin{aligned} 0 &< \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta + \delta_\eta, \mathbf{a}) - \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) \\ &< \delta_E. \end{aligned} \tag{56}$$

Combining the two, we conclude that $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a})$ is a strictly monotonically increasing continuous function of η when $\eta \geqslant 0$ and $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) < 0$. Taking $\eta = 0$, we have $\min_{\mathbf{a}\neq\mathbf{0}} E'(0, \mathbf{a}) = E_g + \epsilon_K$. Therefore, when $\epsilon > \epsilon_K$ and $E_g + \epsilon < 0$, the equation $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) = E_g + \epsilon$ has a unique solution. □

C.1 Proof of Lemma 1

Proof. For a regularisation parameter η satisfying Eq. (6), the first condition of Lemma 7 holds up to the failure probability κ. Under conditions $E_g < 0$ and $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a}) < 0$, according to Lemma 7, $E_g \leqslant \hat{E}_{min} \leqslant \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta, \mathbf{a})$ holds up to the failure probability κ. Eq. (8) is proven. □

C.2 Proof of Theorem 1

Proof. Suppose $\epsilon > \epsilon_K$ and $E_g + \epsilon < 0$. The existence of the solution η is proved in Lemma 8. With the solution η, Eq. (9) is satisfied.

If we take the measurement number

$$M = \frac{4d^4}{\kappa\eta^2}, \tag{57}$$

$\|\hat{\mathbf{H}} - \mathbf{H}\|_2 \leqslant C_{\mathbf{H}}\eta$ and $\|\hat{\mathbf{S}} - \mathbf{S}\|_2 \leqslant C_{\mathbf{S}}\eta$ hold up to the failure probability κ as proved in

Lemma 2. Then, according to Lemma 7, $E_g \leqslant \min_{\mathbf{a}\neq\mathbf{0}} \hat{E}(\eta,\mathbf{a}) \leqslant \min_{\mathbf{a}\neq\mathbf{0}} E'(\eta,\mathbf{a}) = E_g+\epsilon$ holds up to the failure probability.

In the independent measurement protocol, the total number of measurements is

$$M_{tot} = M\times 2\times d^2\times 2 = \frac{16d^6}{\kappa\eta^2}. \quad (58)$$

Notice that for each matrix entry $\hat{\mathbf{H}}_{k,q}$ or $\hat{\mathbf{S}}_{k,q}$, we need M measurements for its real part and M measurements for its imaginary part. There are two matrices, and each matrix has d^2 entries. To match Eq. (58), we can take $\alpha(\kappa) = 256/\kappa$ and $\beta(d) = d^6$ in Eq. (10). □

In the above proof, we have used Lemma 2 to give the most general and conservative estimation of the measurement cost. In fact, according to Appendix B, when considering details of the quantum KSD algorithms, M (and M_{tot}) can be optimised, so as $\alpha(\kappa)$ and $\beta(d)$; see Appendix E.

C.3 Remarks on the measurement number

First, in our algorithm (i.e. GP) and some other KSD algorithms (i.e. P, CP, IP, ITE and F), matrices $\mathbf{H}$ and $\mathbf{S}$ are real. In this case, we only need to measure the real part. The cost is directly reduced by a factor of two. Only measuring the real part also reduces variances by a factor of two, i.e. from $\mathrm{Var}(\hat{\mathbf{H}}_{k,q}) \leqslant 2C_{\mathbf{H}}^2/M$ and $\mathrm{Var}(\hat{\mathbf{S}}_{k,q}) \leqslant 2C_{\mathbf{S}}^2/M$ to $\mathrm{Var}(\hat{\mathbf{H}}_{k,q}) \leqslant C_{\mathbf{H}}^2/M$ and $\mathrm{Var}(\hat{\mathbf{S}}_{k,q}) \leqslant C_{\mathbf{S}}^2/M$. Because of the reduced variance, the cost is reduced by another factor of two. Overall, the measurement cost is reduced by a factor of four in algorithms using real matrices.

Second, when the failure probability κ is low, typically $\|\hat{\mathbf{H}}-\mathbf{H}\|_2 \ll C_{\mathbf{H}}\eta$ and $\|\hat{\mathbf{S}}-\mathbf{S}\|_2 \ll C_{\mathbf{S}}\eta$. In this case, the error is overestimated in E', and the typical error is approximately given by

$$E''(\eta,\mathbf{a}) = \frac{\mathbf{a}^\dagger(\mathbf{H}+C_{\mathbf{H}}\eta)\mathbf{a}}{\mathbf{a}^\dagger(\mathbf{S}+C_{\mathbf{S}}\eta)\mathbf{a}}. \quad (59)$$

Notice that a factor of two has been removed from the denominator and numerator compared with E'. If we consider the typical error $\epsilon = E''-E_g$ instead of the error upper bound $\epsilon = E'-E_g$, the required sampling cost is reduced by a factor of four.

D Cost for measuring the energy with an ideal projection

We consider the ideal case that we can realise an ideal projection onto the true ground state. To use the above results, we assume that $f_1 = |\psi_g\rangle\langle\psi_g|$ is the projection operator and take $d=1$ in the KSD algorithm. Then, we have

$$\mathbf{H}_{1,1} = \langle\varphi|f_1^\dagger H f_1|\varphi\rangle = p_g E_g, \quad (60)$$
$$\mathbf{S}_{1,1} = \langle\varphi|f_1^\dagger f_1|\varphi\rangle = p_g, \quad (61)$$

and $E_{min} = \mathbf{H}_{1,1}/\mathbf{S}_{1,1} = E_g$, i.e. $\epsilon_K = 0$.

We suppose $C_{\mathbf{H}} = \|H\|_2$ and $C_{\mathbf{S}} = 1$. When we take $\eta = \frac{p_g\epsilon}{4\|H\|_2}$, $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta,\mathbf{a}) \leqslant E_g+\epsilon$. Notice that $\min_{\mathbf{a}\neq\mathbf{0}} E'(\eta,\mathbf{a}) = (\mathbf{H}_{1,1}+2\|H\|_2\eta)/(\mathbf{S}_{1,1}+2\eta) \leqslant E_g + 4\|H\|_2\eta/p_g$. Accordingly, the total measurement number in the ideal case has an upper bound

$$M_{tot} = \frac{\alpha(\kappa)\|H\|_2^2}{p_g^2\epsilon^2}, \quad (62)$$

which is the first factor in Eq. (11).

E Optimised α and β factors

Theorem 1 is proved taking the bound given by Lemma 2. However, Lemma 2 aims to give a general result, leading to a conservative estimation of α and β. In this section, based on other results in Appendix B, we will show that these two factors can be further reduced. We summarise the reduced α and β in Table 2 as well.

For the independent measurement protocol, Lemma 3 guarantees that the lower bound of the total measurement number is

$$M_{tot} = 4d^2M = \frac{8d^4}{\eta^2}\ln\frac{8d^2}{\kappa}. \quad (63)$$

Assuming $\kappa \ll \frac{1}{8d^2}$ and neglecting $8d^2$ in logarithm yields $\alpha \approx 128\ln\frac{1}{\kappa}$ and $\beta \approx d^4$.

Definition 3. Let $\mathbf{g} \in \mathbb{R}^{2d-1}$ and $\mathbf{G} \in \mathbb{R}^{d\times d}$. If $\mathbf{G}_{i,j} = \mathbf{g}_{i+j-1}$, then $\mathbf{G}$ is a $d\times d$ real Hankel matrix.

In our algorithm (i.e. GP) and some other KSD algorithms (i.e. P, IP and ITE), $\mathbf{H}$ and $\mathbf{S}$ are real Hankel matrices. The collective measurement protocol only requires measuring $2d-1$ matrix entries for each matrix to construct $\hat{\mathbf{H}}$

and $\hat{\mathbf{S}}$. Specifically, we measure $\mathbf{H}_{\lceil \frac{l}{2} \rceil, \lfloor \frac{l}{2} \rfloor}$ and $\mathbf{S}_{\lceil \frac{l}{2} \rceil, \lfloor \frac{l}{2} \rfloor}$ with $l = 2, 3, \ldots, 2d$. Then we take $\mathbf{H}_{i,j} = \mathbf{H}_{\lceil \frac{i+j}{2} \rceil, \lfloor \frac{i+j}{2} \rfloor}$ and $\mathbf{S}_{i,j} = \mathbf{S}_{\lceil \frac{i+j}{2} \rceil, \lfloor \frac{i+j}{2} \rfloor}$ for all i and j. According to Lemma 4, the total measurement number is

$$\begin{aligned} M_{tot} &= M \times (2d-1) \times 2 \\ &= \frac{4d(2d-1)}{\eta^2} \ln \frac{4d}{\kappa}. \end{aligned} \tag{64}$$

Assuming $\kappa \ll \frac{1}{4d}$ and neglecting $4d$ in logarithm yields $\alpha = 64 \ln \frac{1}{\kappa}$ and $\beta = d(2d-1)$.

For the CP and F algorithms, $\mathbf{H}$ and $\mathbf{S}$ are real symmetric matrices. We need to measure $d(d+1)/2$ matrix entries to construct $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$, respectively. According to Lemma 4, the total measurement number is

$$\begin{aligned} M_{tot} &= M \times d(d+1)/2 \times 2 \\ &= \frac{2d^2(d+1)}{\eta^2} \ln \frac{4d}{\kappa}. \end{aligned} \tag{65}$$

Assuming $\kappa \ll \frac{1}{4d}$ and neglecting $4d$ in logarithm yields $\alpha = 32 \ln \frac{1}{\kappa}$ and $\beta = d^2(d+1)$.

The RTE algorithm is the only one having complex matrices $\mathbf{H}$ and $\mathbf{S}$. Considering its Hermitian–Toeplitz structure, we need to measure d real and $d-1$ imaginary matrix entries to estimate $\mathbf{H}$ and $\mathbf{S}$, respectively. According to Lemma 5, the total measurement number is

$$\begin{aligned} M_{tot} &= M \times (2d-1) \times 2 \\ &= \frac{4(2d-1)^2}{\eta^2} \ln \frac{4d}{\kappa}. \end{aligned} \tag{66}$$

Assuming $\kappa \ll \frac{1}{4d}$ and neglecting $4d$ in logarithm yields $\alpha = 64 \ln \frac{1}{\kappa}$ and $\beta = (2d-1)^2$.

Finally, the optimised α and β are listed in Table 2. Compared with cases, Hankel matrices have smaller α and β. For the GP algorithm, taking $\alpha = 16 \ln \frac{1}{\kappa}$ and $\beta = d(2d-1)$ is sufficient.

F Gaussian-power basis

In this section, we prove Eq. (14) and Eq. (15), respectively.

F.1 Norm upper bound

Lemma 9. *The Gaussian-power basis operators f_k defined in Eq. (13) satisfy Eq. (14).*

Proof. By the spectral decomposition of the operator $(H - E_0)\tau$, we have

$$\tau^{k-1} \|f_k\|_2 \leqslant \max_{y \in \mathbb{R}} |y^{k-1} e^{-\frac{1}{2}y^2}|, \tag{67}$$

where y corresponds to eigenvalues of $(H - E_0)\tau$. It can be found that

$$\max_{y \in \mathbb{R}} |y^{k-1} e^{-\frac{1}{2}y^2}| = \left(\frac{k-1}{e} \right)^{\frac{k-1}{2}}. \tag{68}$$

Combining Eqs. (67) and (68), Eq. (14) is proved. □

F.2 The integral

The generating function for the Hermite polynomials is [40]

$$\sum_{n=0}^{\infty} H_n(u) \frac{t^n}{n!} = e^{2ut - t^2}. \tag{69}$$

Lemma 10.

$$\int_{-\infty}^{+\infty} du H_n(u) e^{-u^2 - iyu} = \sqrt{\pi}(-iy)^n e^{-\frac{1}{4}y^2} \tag{70}$$

holds for arbitrary $y \in \mathbb{R}$.

Proof. Denote $I_n = \int_{-\infty}^{+\infty} du H_n(u) e^{-u^2 - iyu}$. Taking Eq. (69) into account, we have [58]

$$\begin{aligned} \sum_{n=0}^{\infty} I_n \frac{t^n}{n!} &= \int_{-\infty}^{+\infty} du e^{-u^2 + (2t - iy)u - t^2} \\ &= \sqrt{\pi} e^{-\frac{1}{4}y^2 - iyt} \\ &= \sqrt{\pi} e^{-\frac{1}{4}y^2} \sum_{n=0}^{\infty} (-iy)^n \frac{t^n}{n!}, \end{aligned} \tag{71}$$

where the Gaussian integral is used. From Eq. (71), we conclude that $I_n = \sqrt{\pi}(-iy)^n e^{-\frac{1}{4}y^2}$. □

Lemma 11. *Eq. (15) is true.*

Proof. Taking $y = \sqrt{2}x\tau$ and $u = \frac{t}{\sqrt{2}\tau}$ in Eq. (70), Eq. (15) is proved directly. □

G Algorithm and variance

In this section, first, we review the zeroth-order leading-order-rotation formula [47], then we analyse the variance and work out the upper bounds of the cost, finally, we give the pseudocode of our measurement-efficient algorithm.

G.1 Zeroth-order leading-order-rotation formula

Assume that the Hamiltonian is expressed as a linear combination of Pauli operators, i.e.

$$H = \sum_j h_j \sigma_j. \tag{72}$$

The Taylor expansion of the time evolution operator is

$$e^{-iH\Delta t} = \mathbb{1} - iH\Delta t + T_0(\Delta t), \tag{73}$$

where the summation of high-order terms is

$$\begin{aligned} T_0(\Delta t) &= \sum_{k=2}^{\infty} \sum_{j_1,\ldots,j_k} \frac{\prod_{a=1}^{k}(-ih_{j_a}\Delta t)}{k!} \\ &\quad \times \sigma_{j_k} \cdots \sigma_{j_1}. \end{aligned} \tag{74}$$

The leading-order terms can be expressed as a linear combination of rotation operators,

$$\mathbb{1} - iH\Delta t = \sum_j \beta_j(\Delta t) e^{-i\mathrm{sgn}(h_j)\phi(\Delta t)\sigma_j}, \tag{75}$$

where $\phi(\Delta t) = \arctan(h_{tot}\Delta t)$, $\beta_j(\Delta t) = |h_j|\Delta t/\sin\phi(\Delta t)$ and $h_{tot} = \sum_j |h_j|$. The zeroth-order leading-order-rotation formula is

$$\begin{aligned} e^{-iH\Delta t} &= \sum_j \beta_j(\Delta t) e^{-i\mathrm{sgn}(h_j)\phi(\Delta t)\sigma_j} \\ &\quad + T_0(\Delta t), \end{aligned} \tag{76}$$

which is a linear combination of rotation and Pauli operators. Accordingly, for the evolution time t and time step number N, the LCU expression of the time evolution operator is

$$\begin{aligned} e^{-iHt} &= \left[\sum_j \beta_j(t/N) e^{-i\mathrm{sgn}(h_j)\phi(t/N)\sigma_j}\right. \\ &\quad \left. + T_0(t/N)\right]^N. \end{aligned} \tag{77}$$

The above equation is the explicit form of the expression $e^{-iHt} = [\sum_r v_r(t/N)V_r(t/N)]^N$ referred in the main text.

Now, we consider the cost factor, i.e. the 1-norm of coefficients in an LCU expression. For Eq. (76), the corresponding cost factor is

$$\begin{aligned} c(\Delta t) &= \sum_j |\beta_j(\Delta t)| + \sum_{k=2}^{\infty} \sum_{j_1,\ldots,j_k} \frac{\prod_{a=1}^{k} |h_{j_a}\Delta t|}{k!} \\ &= \sqrt{1 + h_{tot}^2\Delta t^2} + e^{h_{tot}|\Delta t|} - (1 + h_{tot}|\Delta t|) \end{aligned} \tag{78}$$

Accordingly, the cost factor of Eq. (77) is $[c(t/N)]^N$.

Lemma 12.

$$c(\Delta t) \leqslant e^{\frac{e}{2}h_{tot}^2\Delta t^2}. \tag{79}$$

Proof. Let $x = h_{tot}|\Delta t|$, thus $x \geqslant 0$. Then

$$\begin{aligned} c(\Delta t) &= \sqrt{1+x^2} + e^x - (1+x) \\ &\leqslant e^{\frac{x^2}{2}} + e^x - (1+x). \end{aligned} \tag{80}$$

Define the function

$$y(x) = e^{\frac{e}{2}x^2} + (1+x) - (e^x + e^{\frac{x^2}{2}}). \tag{81}$$

It follows that $y(0) = 0$ and

$$\begin{aligned} y'(x) &= exe^{\frac{e}{2}x^2} + 1 - (e^x + xe^{\frac{x^2}{2}}) \\ &\geqslant (e-1)xe^{\frac{e}{2}x^2} + 1 - e^x \qquad (82) \\ &\geqslant (e-1)x + 1 - e^x. \qquad (83) \end{aligned}$$

Let $z(x) = (e-1)x+1-e^x$. Since $z(0) = z(1) = 0$, it is easy to see that $z(x) \geqslant 0$ when $0 \leqslant x \leqslant 1$, which indicates that $y'(x) \geqslant 0$ when $0 \leqslant x \leqslant 1$. Based on Eq. (82) and the fact that $e-1 > 1$, we have $y'(x) \geqslant 1$ when $x \geqslant 1$. As a result, $y'(x) \geqslant 0$ when $x \geqslant 0$, which means $y(x) \geqslant 0$. Therefore,

$$c(\Delta t) \leqslant e^{\frac{e}{2}x^2} - y(x) \leqslant e^{\frac{e}{2}x^2}. \tag{84}$$

□

According to Lemma 12, the cost factor e^{-iHt} has the upper bound

$$[c(t/N)]^N \leqslant e^{\frac{eh_{tot}^2t^2}{2N}}. \tag{85}$$

Therefore, the cost factor approaches one when the time step number N increases.

G.2 Variance analysis

In this section, first, we prove the general upper bound of the variance, then we work out the upper bound of the cost c_k.

G.2.1 Variance upper bound

Given the LCU expression $A = \sum_s q_s U_s$, we can compute $\langle\varphi|A|\varphi\rangle$ using the Monte Carlo method and the one-ancilla Hadamard-test circuit shown in Fig. 3. In the Monte Carlo calculation, we sample the unitary operator U_s with a probability of $|q_s|/C_A$ in the spirit of importance sampling. The corresponding algorithm is given in Algorithm 2.

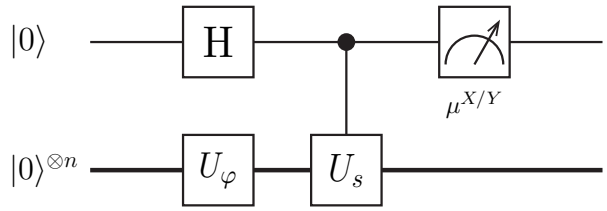


Figure 3: The Hadamard-test circuit $\mathcal{C}_s$. The qubit on the top is the ancilla qubit. The unitary operator U_φ prepares the state $|\varphi\rangle$, i.e. $U_\varphi|0\rangle^{\otimes n} = |\varphi\rangle$. When the ancilla qubit is measured in the X (or Y) basis, the measurement outcome is an eigenvalue of the Pauli operator X (or Y), i.e. $\mu^X = \pm 1$ (or $\mu^Y = \pm 1$).

Algorithm 2 Monte Carlo evaluation of an LCU expression of A.

1: Input $|\varphi\rangle$, $\{(q_s, U_s)\}$ and M.
2: $C_A \leftarrow \sum_s |q_s|$
3: $\hat{A} \leftarrow 0$
4: **for** $l = 1$ to M **do**
5: Choose s with a probability of $|q_s|/C_A$.
6: Implement the circuit $\mathcal{C}_s$ for one shot, measure the ancilla qubit in the X basis and record the measurement outcome μ^X.
7: Implement the circuit $\mathcal{C}_s$ for one shot, measure the ancilla qubit in the Y basis and record the measurement outcome μ^Y.
8: $\hat{A} \leftarrow \hat{A} + e^{i\arg(q_s)}(\mu^X + i\mu^Y)$
9: Output $\hat{A} \leftarrow \frac{C_A}{M}\hat{A}$.

Lemma 13. *According to Algorithm 2, the estimator $\hat{A}$ is unbiased, and the variance upper bound in Eq. (18) is true.*

Proof. First, we rewrite the LCU expression as

$$A = C_A \sum_s \frac{|q_s|}{C_A} e^{i\arg(q_s)} U_s. \tag{86}$$

Then,

$$\langle\varphi|A|\varphi\rangle = C_A \sum_s \frac{|q_s|}{C_A} e^{i\arg(q_s)} \langle\varphi|U_s|\varphi\rangle. \tag{87}$$

Each unitary operator U_s has a corresponding Hadamard-test circuit denoted by $\mathcal{C}_s$, which is shown in Fig. 3. When the ancilla qubit is measured in the $W = X, Y$ basis, the measurement has a random outcome $\mu^W = \pm 1$. Let $P^W_{\mu^W}$ be the probability of the measurement outcome μ^W in $\mathcal{C}_s$. According to Ref. [33], we have

$$\begin{aligned}\langle\varphi|U_s|\varphi\rangle &= \sum_{\mu^X=\pm1} P^X_{\mu^X}\mu^X \\ &\quad + i\sum_{\mu^Y=\pm1} P^Y_{\mu^Y}\mu^Y.\end{aligned} \tag{88}$$

The probability of (s, μ^X, μ^Y) is $(|q_s|/C_A)P^X_{\mu^X}P^Y_{\mu^Y}$. Using Eqs. (87) and (88), we have

$$\begin{aligned}\langle\varphi|A|\varphi\rangle &= C_A \sum_{s,\mu^X,\mu^Y} \frac{|q_s|}{C_A} P^X_{\mu^X} P^Y_{\mu^Y} e^{i\arg(q_s)} \\ &\quad \times(\mu^X + i\mu^Y).\end{aligned} \tag{89}$$

The corresponding Monte Carlo algorithm is given in Algorithm 2.

The estimator $\hat{A}$ is unbiased. According to Eq. (89), the expected value of $C_A e^{i\arg(q_s)}(\mu^X + i\mu^Y)$ is $\langle\varphi|A|\varphi\rangle$. Therefore, the expected value of $\hat{A}$ is also $\langle\varphi|A|\varphi\rangle$. Notice that $\hat{A}$ is the average of $C_A e^{i\arg(q_s)}(\mu^X + i\mu^Y)$ over M samples.

The variance of $\hat{A}$ is

$$\begin{aligned}\mathrm{Var}(\hat{A}) &= \frac{1}{M}\left[\sum_{s,\mu^X,\mu^Y} \frac{|q_s|}{C_A} P^X_{\mu^X} P^Y_{\mu^Y}\right. \\ &\quad \left.\times \left|C_A e^{i\arg(q_s)}(\mu^X + i\mu^Y)\right|^2 - |\langle\varphi|A|\varphi\rangle|^2\right] \\ &\leqslant \frac{2C_A^2}{M}.\end{aligned} \tag{90}$$

□

Here, we remark that the summation-form LCU expressions can be generalised to the integral-form LCU expressions, which inherit the same properties of the former.

G.2.2 Cost upper bound

We use a two-level LCU to construct f_k: First, f_k is an integral of RTE operators e^{-iHt}; Second, RTE is realised using the leading-order-rotation method. Substituting Eq. (77) into Eq. (15), we can obtain the eventual LCU expression of f_k, i.e.

$$\begin{aligned}f_k &= \frac{i^{k-1}}{2^{\frac{k-1}{2}}\tau^{k-1}} \int_{-\infty}^{+\infty} dt H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right) g_\tau(t) e^{iE_0 t} \\ &\quad \times \left[\sum_j \beta_j(t/N) e^{-i\mathrm{sgn}(h_j)\phi(t/N)\sigma_j} + T_0(t/N)\right]^N.\end{aligned} \tag{91}$$

Substituting LCU expressions of f_k and f_q as well as the expression of H into $A = f_k^\dagger H f_q, f_k^\dagger f_q$, we can obtain the LCU expression of A. Then, $C_A = h_{tot} c_k c_q, c_k c_q$, respectively, where

$$c_k = \frac{1}{2^{\frac{k-1}{2}}\tau^{k-1}} \int_{-\infty}^{+\infty} dt \left| H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right)\right| g_\tau(t) \times [c(t/N)]^N. \quad (92)$$

Here, we have replaced $\left[\sum_r \left|v_r\left(\frac{t}{N}\right)\right|\right]^N$ with $[c(t/N)]^N$ in Eq. (20).

To work out the cost of A, we have used that the cost factor is additive and multiplicative. Suppose LCU expressions of A_1 and A_2 are $A_1 = q_1 U_1 + q_1' U_1'$ and $A_2 = q_2 U_2 + q_2' U_2'$, respectively. The cost factors of A_1 and A_2 are $C_1 = |q_1| + |q_1'|$ and $C_2 = |q_2| + |q_2'|$, respectively. Substituting LCU expressions of A_1 and A_2 into $A = A_1 + A_2$, the expression of A reads $A = q_1 U_1 + q_1' U_1' + q_2 U_2 + q_2' U_2'$. Then, the cost factor of A is $C_A = |q_1| + |q_1'| + |q_2| + |q_2'| = C_1 + C_2$. Substituting LCU expressions of A_1 and A_2 into $A' = A_1 A_2$, the expression of A' reads $A' = (q_1 U_1 + q_1' U_1')(q_2 U_2 + q_2' U_2') = q_1 q_2 U_1 U_2 + q_1 q_2' U_1 U_2' + q_1' q_2 U_1' U_2 + q_1' q_2' U_1' U_2'$. Then, the cost factor of A' is $C_A' = |q_1 q_2| + |q_1 q_2'| + |q_1' q_2| + |q_1' q_2'| = C_1 C_2$.

The upper bound of c_k is

$$c_k \leqslant c_k^{ub} = \frac{1}{2^{\frac{k-1}{2}}\tau^{k-1}} \int_{-\infty}^{+\infty} dt \left| H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right)\right| g_\tau(t) \times e^{\chi \frac{t^2}{\tau^2}}, \quad (93)$$

where $\chi = \frac{e h_{tot}^2 \tau^2}{2N}$. Here, we have used Lemma 12. Notice that c_k^{ub} increases monotonically with χ (i.e. decreases monotonically with N). Taking $\chi = \frac{1}{8}$ (i.e. $N = 4e h_{tot}^2 \tau^2$), we numerically evaluate the upper bound c_k^{ub} and plot it in Fig. 4. We can find that

$$c_k^{ub} \leqslant 2\left(\frac{k-1}{e\tau^2}\right)^{\frac{k-1}{2}}. \quad (94)$$

G.3 Pseudocode

Given the LCU expression of A, we can compute $\langle\varphi|A|\varphi\rangle$ according to Algorithm 2. We would like to repeat how we compose the LCU expression of A: Substituting Eq. (77) into Eq. (15), we can obtain the eventual LCU expression of f_k; Substituting LCU expressions of f_k and f_q as well as the expression of H into $A = f_k^\dagger H f_q$ and $A = f_k^\dagger f_q$ (corresponding to $\mathbf{H}_{k,q}$ and $\mathbf{S}_{k,q}$, respectively), we can obtain the LCU expression of A. Algorithm 2 does not involve details of the LCU expression. In this section, we give pseudocodes involving details.

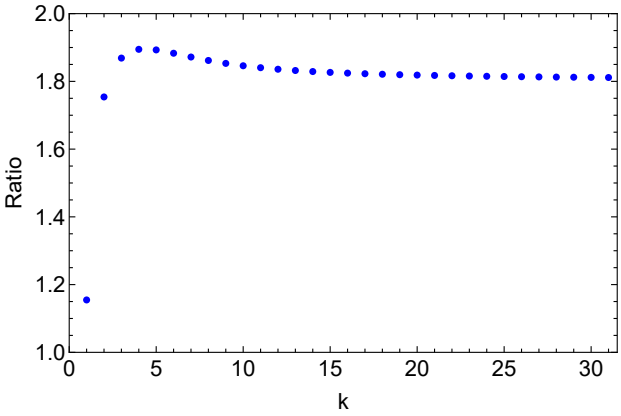


Figure 4: The ratio between cost upper bound and norm upper bound, $c_k^{ub}/[(k-1)/(e\tau^2)]^{(k-1)/2}$.

We remark that matrix entries of the rescaled basis $f_k' = f_k/c_k$ are $\hat{\mathbf{H}}'_{k,q} = \hat{\mathbf{H}}_{k,q}/(c_k c_q)$ and $\hat{\mathbf{S}}'_{k,q} = \hat{\mathbf{S}}_{k,q}/(c_k c_q)$, where $\hat{\mathbf{H}}_{k,q}$ and $\hat{\mathbf{S}}_{k,q}$ are computed according to the pseudocodes.

In pseudocodes, we use notations $\mathbf{h} = (h_1, h_2, \ldots)$ and $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots)$ to represent the Hamiltonian $H = \sum_j h_j \sigma_j$. We define probability density functions

$$p_{\tau,k}(t) = \frac{1}{c_k 2^{\frac{k-1}{2}}\tau^{k-1}} \left| H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right)\right| g_\tau(t) \times [c(t/N)]^N, \quad (95)$$

and probabilities

$$P_{\mathrm{L}}(\Delta t) = \frac{\sqrt{1 + h_{tot}^2 \Delta t^2}}{c(\Delta t)}, \quad (96)$$

$$P_{\mathrm{T}}(\Delta t) = \frac{e^{h_{tot}|\Delta t|} - (1 + h_{tot}|\Delta t|)}{c(\Delta t)}. \quad (97)$$

We use $\textsc{Poisson}(\lambda)$ to denote a subroutine that returns k with a probability of $\lambda^k e^{-\lambda}/k!$. The LCU expression of f_k is sampled according to Algorithms 5, 6 and 7. In Algorithm 5, we generate random time t according to the distribution $p_{\tau,k}(t)$. In Algorithm 6, we realise the RTE e^{-iHt} by N steps leading-order-rotation. In Algorithm 7, we randomly sample rotations or Pauli operators according to Eq. (77).

Algorithm 3 Measurement of matrix entry $\mathbf{H}_{k,q}$.

Input $|\varphi\rangle$, $(\mathbf{h},\boldsymbol{\sigma})$, E_0, τ, N and M.
$h_{tot} \leftarrow \sum_j |h_j|$
Compute c_k and c_q.
$\hat{\mathbf{H}}_{k,q} \leftarrow 0$
for $l = 1$ to M **do**
    Choose j with a probability of $|h_j|/h_{tot}$.
    $(\bar{v}_k, \bar{V}_k) \leftarrow$ BasisGen($\mathbf{h}$,$\boldsymbol{\sigma}$,E_0,τ,N,k)
    $(\bar{v}_q, \bar{V}_q) \leftarrow$ BasisGen($\mathbf{h}$,$\boldsymbol{\sigma}$,E_0,τ,N,q)
    Implement the circuit $\mathcal{C}_s$ with $U_s = \bar{V}_k^\dagger \sigma_j \bar{V}_q$ for one shot, measure the ancilla qubit in the X basis and record the measurement outcome μ^X.
    Implement the circuit $\mathcal{C}_s$ with $U_s = \bar{V}_k^\dagger \sigma_j \bar{V}_q$ for one shot, measure the ancilla qubit in the Y basis and record the measurement outcome μ^Y.
    $\hat{\mathbf{H}}_{k,q} \leftarrow \hat{\mathbf{H}}_{k,q} + e^{i\arg(\bar{v}_k^* h_j \bar{v}_q)}(\mu^X + i\mu^Y)$
Output $\hat{\mathbf{H}}_{k,q} \leftarrow \frac{h_{tot} c_k c_q}{M} \hat{\mathbf{H}}_{k,q}$.

Algorithm 4 Measurement of matrix entry $\mathbf{S}_{k,q}$.

Input $|\varphi\rangle$, $(\mathbf{h},\boldsymbol{\sigma})$, E_0, τ, N and M.
Compute c_k and c_q.
$\hat{\mathbf{S}}_{k,q} \leftarrow 0$
for $l = 1$ to M **do**
    $(\bar{v}_k, \bar{V}_k) \leftarrow$ BasisGen($\mathbf{h}$,$\boldsymbol{\sigma}$,E_0,τ,N,k)
    $(\bar{v}_q, \bar{V}_q) \leftarrow$ BasisGen($\mathbf{h}$,$\boldsymbol{\sigma}$,E_0,τ,N,q)
    Implement the circuit $\mathcal{C}_s$ with $U_s = \bar{V}_k^\dagger \bar{V}_q$ for one shot, measure the ancilla qubit in the X basis and record the measurement outcome μ^X.
    Implement the circuit $\mathcal{C}_s$ with $U_s = \bar{V}_k^\dagger \bar{V}_q$ for one shot, measure the ancilla qubit in the Y basis and record the measurement outcome μ^Y.
    $\hat{\mathbf{S}}_{k,q} \leftarrow \hat{\mathbf{S}}_{k,q} + e^{i\arg(\bar{v}_k^* \bar{v}_q)}(\mu^X + i\mu^Y)$
Output $\hat{\mathbf{S}}_{k,q} \leftarrow \frac{c_k c_q}{M} \hat{\mathbf{S}}_{k,q}$.

Algorithm 5 Basis generator f_k.

function BasisGen($\mathbf{h}$,$\boldsymbol{\sigma}$,E_0,τ,N,k)
    Generate random time t according to the distribution $p_{\tau,k}(t)$.
    $(\bar{v}, \bar{V}) \leftarrow$ RTE($\mathbf{h}$,$\boldsymbol{\sigma}$,N,t)
    $\bar{v} \leftarrow \bar{v} \times \frac{i^{k-1}}{2^{\frac{k-1}{2}} \tau^{k-1}} H_{k-1}\left(\frac{t}{\sqrt{2}\tau}\right) g_\tau(t) e^{iE_0 t}$
    Output $(\bar{v}, \bar{V})$.

Algorithm 6 Real-time evolution.

function RTE($\mathbf{h}$,$\boldsymbol{\sigma}$,N,t)
    $\bar{v} \leftarrow 1$
    $\bar{V} \leftarrow \mathbb{1}$
    for $i = 1$ to N **do**
        $(v, V) \leftarrow$ LORF($\mathbf{h}$,$\boldsymbol{\sigma}$,t/N)
        $\bar{v} \leftarrow \bar{v} \times v$
        $\bar{V} \leftarrow \bar{V} \times V$
    Output $(\bar{v}, \bar{V})$.

Algorithm 7 Leading-order-rotation formula.

function LORF($\mathbf{h}$,$\boldsymbol{\sigma}$,Δt)
    Choose O from L and T with probabilities of $P_{\mathrm{L}}(\Delta t)$ and $P_{\mathrm{T}}(\Delta t)$, respectively.
    if O = L **then**
        Choose j with a probability of $|h_j|/h_{tot}$.
        $v \leftarrow \beta_j(\Delta t)$
        $V \leftarrow e^{-i\mathrm{sgn}(h_j)\phi(\Delta t)\sigma_j}$
    else if O = T **then**
        while $k < 2$ **do**
            $k \leftarrow$ Poisson($h_{tot}|\Delta t|$)
        $v \leftarrow 1/k!$
        $V \leftarrow \mathbb{1}$
        for $a = 1$ to k **do**
            Choose j with a probability of $|h_j|/h_{tot}$.
            $v \leftarrow v \times (-ih_j \Delta t)$
            $V \leftarrow V \times \sigma_j$
    Output (v, V).

H Details in numerical calculation

In this section, first, we describe models and reference states taken in the benchmarking, and then we give details about instances and parameters.

H.1 Models and reference states

Two models are used in the benchmarking: the anti-ferromagnetic Heisenberg model

$$H = J \sum_{\langle i,j \rangle} (\sigma_i^X \sigma_j^X + \sigma_i^Y \sigma_j^Y + \sigma_i^Z \sigma_j^Z), \qquad (98)$$

where σ_i^X, σ_i^Y and σ_i^Z are Pauli operators of the ith qubit; and the Hubbard model

$$\begin{aligned} H &= -J\sum_{\langle i,j\rangle}\sum_{\sigma=\uparrow,\downarrow}(a_{i,\sigma}^\dagger a_{j,\sigma}+a_{j,\sigma}^\dagger a_{i,\sigma}) \\ &+ U\sum_i\left(a_{i,\uparrow}^\dagger a_{i,\uparrow}-\frac{\mathbb{1}}{2}\right)\left(a_{i,\downarrow}^\dagger a_{i,\downarrow}-\frac{\mathbb{1}}{2}\right), \end{aligned} \tag{99}$$

where $a_{i,\sigma}$ is the annihilation operator of the ith orbital and spin σ. For the Heisenberg model, we take the spin number as ten, and for the Hubbard model, we take the orbital number as five. Then both models can be encoded into ten qubits. For the Hubbard model, we take $U = J$. Without loss of generality, we normalise the Hamiltonian for simplicity: We take J such that $\|H\|_2 = 1$, i.e. eigenvalues of H are in the interval $[-1, 1]$.

For each model, we consider three types of lattices: chain, ladder and randomly generated graphs, see Fig. 5. We generate a random graph in the following way: i) For a vertex v, randomly choose a vertex $u \neq v$ and add the edge (v, u); ii) Repeat step-i two times for the vertex v; iii) Implement steps i and ii for each vertex v. On a graph generated in this way, each vertex is connected to four vertices on average. Notice that some pairs of vertices are connected by multiple edges, and then the interaction between vertices in the pair is amplified by the number of edges.

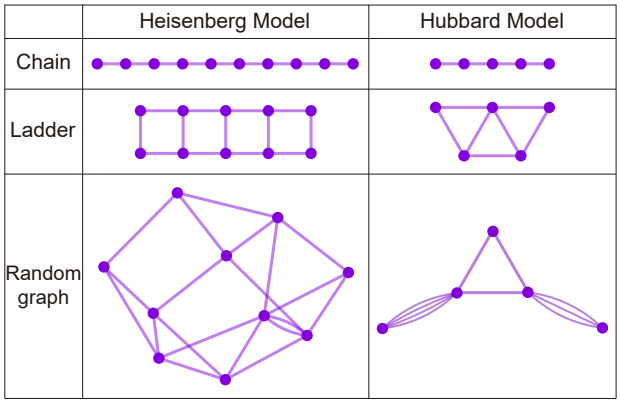


Figure 5: Lattices of the Heisenberg model and Hubbard model. One of the randomly generated graphs for each model is illustrated in the figure as an example.

It is non-trivial to choose and prepare a good reference state that has a sufficiently large overlap with the true ground state. Finding the ground-state energy of a local Hamiltonian is QMA-hard [59, 60]. If one can prepare a good reference state, then quantum computing is capable of solving the ground-state energy using quantum phase estimation [61]. So far, there does not exist a universal method for reference state preparation; see Ref. [62] for relevant discussions. Despite this, there are some practical ways of choosing and preparing the reference state. For fermion systems, we can take the mean-field approximate solution, i.e. a Hartree-Fock state, as the reference state. For general systems, one may try adiabatic state preparation, etc. We stress that preparing good reference states is beyond the scope of this work. Here, we choose two reference states which are likely to overlap with true ground states as examples. For the Heisenberg model, we take the pairwise singlet state

$$|\varphi\rangle = |\Phi\rangle_{1,2}\otimes|\Phi\rangle_{3,4}\otimes\cdots, \tag{100}$$

where

$$|\Phi\rangle_{i,j} = \frac{1}{\sqrt{2}}(|0\rangle_i\otimes|1\rangle_j - |1\rangle_i\otimes|0\rangle_j) \tag{101}$$

is the singlet state of spins i and j. For the Hubbard model, we take a Hartree-Fock state as the reference state: We compute the ground state of the one-particle Hamiltonian (which is equivalent to taking $U = 0$) and take the ground state of the one-particle Hamiltonian (a Slater determinant) as the reference state.

H.2 Instances

We test the algorithms listed in Table 1 with many instances. Each instance is a triple $(model, lattice, d)$, in which *model* takes one of the two models (Heisenberg model and Hubbard model), *lattice* takes a chain, ladder or random graph, and d is the dimension of the subspace.

Instances for computing empirical distributions in Fig. 2 consist of six groups: the Heisenberg model on the chain, ladder and random graphs, and the Hubbard model on the chain, ladder and random graphs. For chain and ladder, we evaluate each subspace dimension $d = 2, 3, \ldots, 30$; For a random graph, we only evaluate one subspace dimension randomly taken in the interval. Some instances are discarded. To avoid any potential issue caused by the precision of floating-point numbers, we discard instances that the subspace error ϵ_K of the P algorithm is lower than 10^{-9} due to a large d. We also discard instances that ϵ_K of the P algorithm is higher than 10^{-2} due to a small d, because the dimension may be too small to achieve an interesting accuracy in

this case. For randomly generated graphs, the two reference states may have a small overlap with the true ground state. Because a good reference state is necessary for all quantum KSD algorithms, we discard graphs with $p_g < 10^{-3}$. Eventually, we have eight instances of the Heisenberg chain, seven instances of the Heisenberg Ladder, twenty-two instances of the Hubbard chain and twenty instances of the Hubbard ladder. For each model, we generate a hundred random-graph instances. The total number of instances is 233.

H.3 Parameters

In this section, we give parameters taken in each algorithm listed in Table 1, namely, E_0, τ, Δt, ΔE, $C_{\mathbf{H}}$ and $C_{\mathbf{S}}$. We summarise these parameters in Table 3.

For E_0, we expect that taking E_0 close to E_g is optimal for our GP algorithm. As the exact value of E_g is unknown, we assume that we have a preliminary estimation of the ground-state energy with uncertainty as large as 10% of the entire spectrum, i.e. we take $E_0 \in [E_g - 0.1, E_g + 0.1]$. For other algorithms, we take E_0 by assuming that the exact value of E_g is known. In the P algorithm, we take $E_0 = E_g + 1$, such that the ground state is the eigenstate of $H - E_0$ with the largest absolute eigenvalue and $\|H - E_0\|_2 = 1$. Similarly, in the IP algorithm, we take $E_0 = E_g - 1$, such that the ground state is the eigenstate of $(H - E_0)^{-1}$ with the largest absolute eigenvalue and $\|(H - E_0)^{-1}\|_2 = 1$. In the ITE algorithm, E_0 causes a constant factor $e^{\tau(k-1)E_0}$ in each operator f_k, i.e. determines the spectral norm of f_k. Because the variance is related to the norm, a large E_0 is problematic. Therefore, in the ITE algorithm, we take $E_0 = E_g$, such that $\|f_k\|_2 = 1$ for all k. In the F algorithm, we expect that $E_0 = E_g$ is optimal because f_1 is an energy filter centred at the ground-state energy in this case. Therefore, we take $E_0 = E_g$ in the F algorithm. The RTE algorithm is closely related to the F algorithm. Therefore, we also take $E_0 = E_g$ in the RTE algorithm.

We remark that in the GP algorithm, we take random E_0 uniformly distributed in the interval $[E_g - 0.1, E_g + 0.1]$ to generate data in Fig. 2, and we take $E_0 = -E_g + i\delta E$, where $i = -50, -9, \ldots, -1, 0, 1, \ldots, 9, 50$ and $\delta E = 0.002$, to generate data in Fig. 1. We remark that we have normalised the Hamiltonian such that $\|H\|_2 = 1$.

For Δt in the F algorithm, we take $\Delta t = 2\tau/(L-1)$. For simplicity, we take the limit $L \to +\infty$, i.e.

$$\begin{aligned} f_k &= \frac{1}{2\tau}\int_{-\tau}^{\tau} dt e^{-i[H-E_0-\Delta E(k-1)]t} \\ &= \frac{\sin\left[H - E_0 - \Delta E\,(k-1)\right]\tau}{\left[H - E_0 - \Delta E\,(k-1)\right]\tau}. \end{aligned} \tag{102}$$

Notice that this f_k is an energy filter centred at $E_0 + \Delta E\,(k-1)$, and the filter is narrower when τ is larger.

Now, we have three algorithms that have the parameter τ: GP, ITE and F. Similar to the F algorithm, f_1 in the GP algorithm is an energy filter centred at E_0. For these two algorithms, if $E_0 = E_g$, $f_1|\varphi\rangle$ converges to the true ground state in the limit $\tau \to +\infty$. It is also similar in the ITE algorithm, in which f_d is a projector onto the ground state, and $f_d|\varphi\rangle$ converges to the true ground state in the limit $\tau \to +\infty$. Realising a large τ is at a certain cost, specifically, the circuit depth increases with τ [42, 8]. Therefore, it is reasonable to take a finite τ. Next, we give the protocol for determining the value of τ in each algorithm.

Without getting into details about realising the filters and projectors, we choose τ under the assumption that if filters and projectors have the same power in computing the ground-state energy, they probably require similar circuit depths. In GP and F algorithms, if $E_0 = E_g$, the energy error achieved by filters reads $\mathbf{H}_{1,1}/\mathbf{S}_{1,1} - E_g$; and in the ITE algorithm, the energy error achieved by the projector reads $\mathbf{H}_{d,d}/\mathbf{S}_{d,d} - E_g$. We take τ such that errors achieved by filters and projectors take the same value ϵ_B. Specifically, for the GP and F algorithms, we determine the value of τ by solving the equation (taking $E_0 = E_g$)

$$\frac{\mathbf{H}_{1,1}(\tau)}{\mathbf{S}_{1,1}(\tau)} - E_g = \epsilon_B; \tag{103}$$

and for the ITE algorithm, we determine the value of τ by solving the equation

$$\frac{\mathbf{H}_{d,d}(\tau)}{\mathbf{S}_{d,d}(\tau)} - E_g = \epsilon_B. \tag{104}$$

To choose the value of ϵ_B, we take the P algorithm as the standard, in which f_d is a projector onto the ground state, i.e. $f_d|\varphi\rangle$ converges to the

Abbr.	E_0	τ	Δt	ΔE	$C_{\mathbf{H}}$	$C_{\mathbf{S}}$
P	E_g+1	N/A	N/A	N/A	1	1
CP	0	N/A	N/A	N/A	1	1
GP	$[E_g-0.1, E_g+0.1]$	Solving Eq. (103)	N/A	N/A	$h_{tot} \geqslant 1$	1
IP	E_g-1	N/A	N/A	N/A	1	1
ITE	E_g	Solving Eq. (104)	N/A	N/A	1	1
RTE	E_g	N/A	Minimising ϵ_K	N/A	1	1
F	E_g	Solving Eq. (103)	$2\tau/(L-1)$	Minimising ϵ_K	1	1

Table 3: Parameters taken in the numerical calculation.

true ground state in the limit $d \to +\infty$. Overall, we determine the value of τ in the following way: Given an instance $(model, lattice, d)$, i) first, compute the energy error achieved by the projector in the P algorithm, take $\epsilon_B = \mathbf{H}_{d,d}/\mathbf{S}_{d,d} - E_g$; then, ii) solve equations of τ. In this way, filters and projectors in P, GP, ITE and F algorithms have the same power.

For Δt in the RTE algorithm and ΔE in the F algorithm, there are works on how to choose them in the literature [14, 17, 15]. In this work, we simply determine their values by a grid search. For the RTE algorithm, we take $\Delta t = i\delta t$, where $i = 1, 2, \ldots, 100$ and $\delta t = \frac{2\pi}{100}$; we compute the subspace error ϵ_K for all Δt; and we choose Δt of the minimum ϵ_K. For the F algorithm, we take $\Delta E = i\delta E$, where $i = 1, 2, \ldots, 100$ and $\delta E = \frac{2}{100d}$ (In this way, when we take the largest ΔE, filters span the entire spectrum); we compute the subspace error ϵ_K for all ΔE; and we choose ΔE of the minimum ϵ_K.

For $C_{\mathbf{H}}$ and $C_{\mathbf{S}}$, we take $C_{\mathbf{H}} = h_{tot}$ and $C_{\mathbf{S}} = 1$ in our GP algorithm following the analysis in Appendix G. For other algorithms, we take these two parameters in the following way. For P, IP, ITE and RTE algorithms, we take $C_{\mathbf{H}}$ and $C_{\mathbf{S}}$ according to spectral norms (the lower bound of cost) without getting into details about measuring matrix entries. In these four algorithms, $\|f_k^\dagger H f_q\|_2 = \|f_k^\dagger f_q\|_2 = 1$ for all k and q, therefore, we take $C_{\mathbf{H}} = C_{\mathbf{S}} = 1$. In the CP algorithm, we take $C_{\mathbf{H}}$ and $C_{\mathbf{S}}$ in a similar way. Because $\|H/h_{tot}\|_2 \leqslant 1$, the spectrum of H/h_{tot} is in the interval $[-1, 1]$. In this interval, the Chebyshev polynomial of the first kind takes values in the interval $[-1, 1]$. Therefore, $\|T_k(H/h_{tot})\|_2 \leqslant 1$, and consequently $\|f_k^\dagger H f_q\|_2, \|f_k^\dagger f_q\|_2 \leqslant 1$. These norms depend on the spectrum of H. For simplicity, we take the upper bound of norms, i.e. $C_{\mathbf{H}} = C_{\mathbf{S}} = 1$, in the CP algorithm. In the F algorithm, each f_k is a linear combination of e^{-iHt} operators, i.e. f_k is expressed in the LCU form, and the cost factor of the LCU expression is one. Therefore, we take $C_{\mathbf{H}} = C_{\mathbf{S}} = 1$ in the F algorithm, assuming that H is expressed in the LCU form with a cost factor of one (Notice that one is the lower bound).

I Derivation of Eq. (21)

Define the function $f(\eta) = E'(\eta, \mathbf{a}) = \frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}+2C_{\mathbf{H}}\eta\mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}+2C_{\mathbf{S}}\eta\mathbf{a}^\dagger\mathbf{a}}$. Consider a small η and the Taylor expansion of $f(\eta)$. We have $f(0) = \frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}$ and

$$\begin{aligned} f'(\eta) &= \frac{2\mathbf{a}^\dagger\mathbf{a}(C_{\mathbf{H}}\mathbf{a}^\dagger\mathbf{S}\mathbf{a} - C_{\mathbf{S}}\mathbf{a}^\dagger\mathbf{H}\mathbf{a})}{(\mathbf{a}^\dagger\mathbf{S}\mathbf{a} + 2C_{\mathbf{S}}\eta\mathbf{a}^\dagger\mathbf{a})^2}, \\ f'(0) &= \frac{2\mathbf{a}^\dagger\mathbf{a}(C_{\mathbf{H}}\mathbf{a}^\dagger\mathbf{S}\mathbf{a} - C_{\mathbf{S}}\mathbf{a}^\dagger\mathbf{H}\mathbf{a})}{(\mathbf{a}^\dagger\mathbf{S}\mathbf{a})^2} \\ &= \frac{\mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}} 2\left(C_{\mathbf{H}} - C_{\mathbf{S}}\frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}\right). \end{aligned} \quad (105)$$

The minimum value of the zeroth-order term $\frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}$ is E_{min} cccording to the Rayleigh quotient theorem [51]. Take this minimum value, we have $E' \simeq E_{min} + s\eta$, where $s = f'(0) \propto \frac{\mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}$ is the first derivative. The solution to $E' = E_g + \epsilon$ is

$\eta \simeq (E_g + \epsilon - E_{min})/s$, then

$$\begin{aligned}\gamma &\simeq \frac{\epsilon^2}{16\|H\|_2^2}\frac{4\left(C_{\mathbf{H}} - C_{\mathbf{S}}\frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}\right)^2}{(E_g+\epsilon-E_{min})^2}\left(\frac{p_g\mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}\right)^2 \\ &= u^2\left(\frac{p_g\mathbf{a}^\dagger\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}\right)^2, \end{aligned} \tag{106}$$

where

$$\begin{aligned} u &= \frac{\epsilon\left|C_{\mathbf{H}} - C_{\mathbf{S}}\frac{\mathbf{a}^\dagger\mathbf{H}\mathbf{a}}{\mathbf{a}^\dagger\mathbf{S}\mathbf{a}}\right|}{2\|H\|_2\,|E_g+\epsilon-E_{min}|} \\ &\leqslant \frac{(C_{\mathbf{H}} + \|H\|_2 C_{\mathbf{S}})\epsilon}{2\|H\|_2(\epsilon-\epsilon_K)} \approx C_{\mathbf{S}} \end{aligned} \tag{107}$$

under assumptions $C_{\mathbf{H}} \approx \|H\|_2 C_{\mathbf{S}}$ and $\epsilon - \epsilon_K \approx \epsilon$.

J Composing Chebyshev polynomials

J.1 Chebyshev-polynomial projector

In this section, we explain how to use Chebyshev polynomials as projectors onto the ground state.

The nth Chebyshev polynomial of the first kind is

$$T_n(z) = \begin{cases} \cos(n\arccos z), & \text{if } |z| \leqslant 1, \\ \operatorname{sgn}(z)^n \cosh(n\,\operatorname{arccosh}|z|), & \text{if } |z| > 1. \end{cases} \tag{108}$$

Chebyshev polynomials have the following properties: i) When $|z| \leqslant 1$, $|T_n(z)| \leqslant 1$; and ii) when $|z| > 1$, $|T_n(z)| > (|z|^n + |z|^{-n})/2$ increases exponentially with n.

Let's consider the spectral decomposition of the Hamiltonian, $H = \sum_{m=1}^{d_{\mathcal{H}}} E_m|\psi_m\rangle\langle\psi_m|$. Here, $d_{\mathcal{H}}$ is the dimension of the Hilbert space, E_m are eigenenergies of the Hamiltonian, and $|\psi_m\rangle$ are eigenstates. We suppose that eigenenergies are sorted in ascending order, i.e. $E_i \leqslant E_j$ if $i < j$. Then, $E_1 = E_g$ is the ground-state energy (accordingly, $|\psi_1\rangle = |\psi_g\rangle$ is the ground state) and E_2 is the energy of the first excited state. The energy gap between the ground state and the first excited state is $\Delta = E_2 - E_1$.

To compose a projector, we take

$$Z = 1 - \frac{H - E_2}{\|H\|_2}. \tag{109}$$

Notice that Z and H have the same eigenstates. The spectral decomposition of Z is $Z = \sum_{m=1}^{d_{\mathcal{H}}} z_m|\psi_m\rangle\langle\psi_m|$, where

$$z_m = 1 - \frac{E_m - E_2}{\|H\|_2}. \tag{110}$$

Then, the ground state corresponds to $z_1 = 1 + \frac{\Delta}{\|H\|_2} > 1$ (suppose the gap is finite), and the first excited state corresponds to $z_2 = 1$. Because $|E_m - E_2| \leqslant 2\|H\|_2$, $z_m \geqslant -1$ for all m. Therefore, except the ground state, z_m of all excited states (i.e. $m \geqslant 2$) is in the interval $[-1, 1]$.

The projector reads

$$\begin{aligned}\frac{T_n(Z)}{T_n(z_1)} &= \sum_{m=1}^{d_{\mathcal{H}}}\frac{T_n(z_m)}{T_n(z_1)}|\psi_m\rangle\langle\psi_m| \\ &= |\psi_g\rangle\langle\psi_g| + \Omega, \end{aligned} \tag{111}$$

where

$$\Omega = \sum_{m=2}^{d_{\mathcal{H}}}\frac{T_n(z_m)}{T_n(z_1)}|\psi_m\rangle\langle\psi_m|. \tag{112}$$

Ω and H have the same eigenstates. Because $|T_n(z_m)| \leqslant 1$ when $m \geqslant 2$,

$$\|\Omega\|_2 \leqslant \frac{1}{T_n(z_1)}. \tag{113}$$

The key in using Chebyshev polynomials as projectors is laying all excited states in the interval $z \in [-1, 1]$ and leaving the ground state out of the interval. For the CP basis, all eigenstates are in the interval $z \in [-1, 1]$. Therefore, f_k operators of the CP basis are not projectors in general.

J.2 Expanding the projector as Hamiltonian powers

In this section, we expand the Chebyshev-polynomial projector as a linear combination of Hamiltonian powers $(H - E_0)^{k-1}$. We focus on the case that $E_0 = E_g$.

The explicit expression of $T_n(z)$ $(n > 0)$ is

$$\begin{aligned} T_n(z) &= n\sum_{m=0}^{n}(-2)^m\frac{(n+m-1)!}{(n-m)!(2m)!} \\ &\quad\times(1-z)^m. \end{aligned} \tag{114}$$

Because

$$\begin{aligned}(1-Z)^m &= \sum_{l=0}^{m}\frac{m!}{(m-l)!l!} \\ &\quad\times\frac{(E_0-E_2)^{m-l}(H-E_0)^l}{\|H\|_2^m}, \end{aligned} \tag{115}$$

we have

$$T_n(Z) = \sum_{l=0}^{n} b_l \frac{(H-E_0)^l}{\|H\|_2^l}, \tag{116}$$

where

$$\begin{aligned} b_l &= n\sum_{m=l}^{n}(-2)^m \frac{(n+m-1)!}{(n-m)!(2m)!}\frac{m!}{(m-l)!l!} \\ &\times \frac{(E_0-E_2)^{m-l}}{\|H\|_2^{m-l}}. \end{aligned} \tag{117}$$

Now, we give an upper bound of $|b_l|$. First, we consider $l=0$. In this case,

$$\begin{aligned} |b_0| &\leqslant n\sum_{m=0}^{n} 2^m \frac{(n+m-1)!}{(n-m)!(2m)!}\frac{|E_0-E_2|^m}{\|H\|_2^m} \\ &= T_n(z_1). \end{aligned} \tag{118}$$

Then, for an arbitrary l,

$$|b_l| \leqslant n^l T_n(z_1), \tag{119}$$

where the inequality

$$\frac{m!}{(m-l)!l!} \leqslant m^l \leqslant n^l \tag{120}$$

has been used.

Finally, taking $d = n+1$ and

$$a_k = \frac{c_k b_{k-1}}{T_n(z_1)\|H\|_2^{k-1}}, \tag{121}$$

we have

$$\sum_{k=1}^{d} a_k c_k^{-1} f_k = \frac{T_n(Z)}{T_n(z_1)} e^{-\frac{1}{2}(H-E_0)^2\tau^2}, \tag{122}$$

where f_k are operators defined in Eq. (13). Because of the upper bound

$$\begin{aligned} |a_k| &\leqslant 2\left(\frac{k-1}{e\tau^2}\right)^{\frac{k-1}{2}}\left(\frac{n}{\|H\|_2}\right)^{k-1} \\ &\leqslant 2\left(\frac{n^3}{e\|H\|_2^2\tau^2}\right)^{\frac{k-1}{2}}, \end{aligned} \tag{123}$$

the overhead factor is

$$\begin{aligned} \gamma &\approx \left(\mathbf{a}^\dagger\mathbf{a}\right)^2 \leqslant 4\sum_{k=1}^{d}\left(\frac{n^3}{e\|H\|_2^2\tau^2}\right)^{k-1} \\ &\leqslant 4\frac{1}{1-n^3/(e\|H\|_2^2\tau^2)}. \end{aligned} \tag{124}$$

K The choice of τ

There are two bounds determining the range of τ. First, we take $\tau > \sqrt{\frac{d-1}{e}}$, such that the upper bound of $\|f_k\|_2$ decrease exponentially with k. Second, under the assumption $|E_0 - E_g| \leqslant \epsilon_0$, we take $\tau \leqslant \frac{\sqrt{d-1}}{\epsilon_0}$. To justify the second bound, we plot the absolute value of the rescaled Gaussian-power function $|f_k'(x)| = |f_k(x)|/c_k$ with different k in Fig. 6, where the instance $(\text{Heisenberg}, \text{chain}, d = 5)$ is taken as an example. This function has maximum points at $\left(\pm\frac{\sqrt{k-1}}{\tau}, (\frac{k-1}{e\tau^2})^{\frac{k-1}{2}}/c_k\right)$. When k increases from 1 to d, the peaks of $f_k'(x)$ span in the range $\left[-\frac{\sqrt{d-1}}{\tau}, \frac{\sqrt{d-1}}{\tau}\right]$, making the basis a filter [2]. It is preferred that the ground-state energy is in the range, which leads to the second bound. Notice that we can always normalise the Hamiltonian by taking $H \leftarrow H/h_{tot}$. Then, $\|H\|_2 \leqslant 1$, $|E_0 - E_g| \leqslant 1$ for all $E_0 \in [-1, 0]$, and the two bounds are consistent.

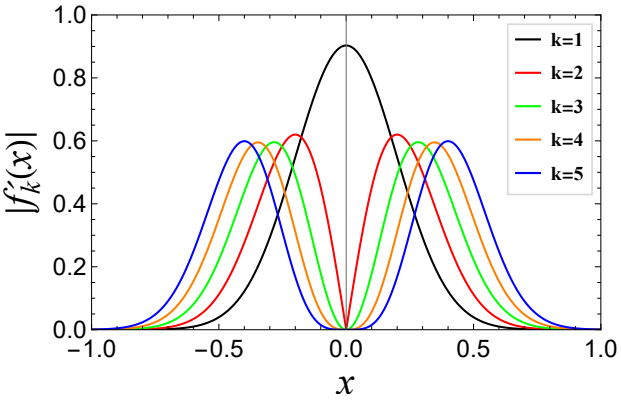


Figure 6: The absolute value of the rescaled Gaussian-power function $|f_k'(x)| = |x^{k-1}e^{-x^2\tau^2/2}|/c_k$ with $k = 1$ to d and $\tau = 5$. Here, we take the instance $(\text{Heisenberg}, \text{chain}, d = 5)$ as an example.

L Numerical results of the measurement number

In Theorem 1, we give a theoretical result about the sufficient measurement number, which is used in later numerical study. In this section, we numerically calculate the necessary measurement number and compare the regularisation method to the thresholding method.

L.1 Necessary measurement costs

Due to the statistical error, the output energy $\hat{E}_{min}$ follows a certain distribution in the vicinity of the true ground-state energy E_g. Given the permissible energy error ϵ and failure probability κ, a measurement number is said to be sufficient if $\hat{E}_{min}$ is in the range $[E_g - \epsilon, E_g + \epsilon]$ with a probability not lower than $1 - \kappa$; and the necessary measurement number is the minimum sufficient measurement number.

Taking the instance $(\text{Heisenberg}, \text{chain}, d = 5)$ as an example, Fig. 7 shows the errors in the ground-state energy and the corresponding necessary/sufficient measurement numbers. We can find that the sufficient measurement costs are close to the necessary measurement numbers.

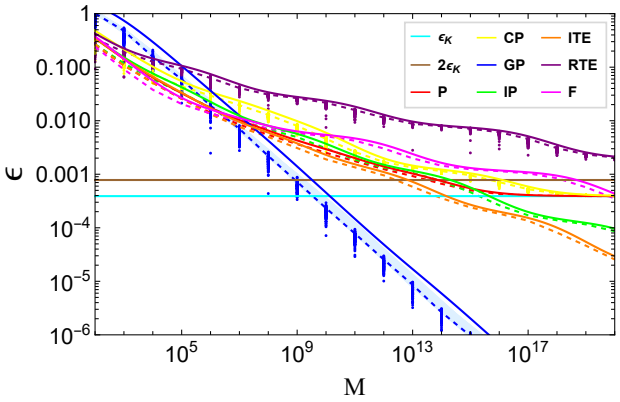


Figure 7: The variational ground-state energy errors ϵ and the corresponding measurement numbers M for the quantum KSD algorithms listed in Table 1. The instance is $(\text{Heisenberg}, \text{chain}, d = 5)$ and $E_g = -1$. The solid curves indicate the sufficient measurement numbers computed through Theorem 1. The dots denote errors at various measurement numbers computed following Algorithm 1. At each measurement number M, Algorithm 1 is implemented for 100 times (simulated on a classical computer), and $(M, |\hat{E}_{min} - E_g|)$ of each implementation is plotted in the figure. The dashed lines are taken such that 10% dots are above them, i.e. the dashed lines corresponding to the necessary measurement number for the failure probability $\kappa = 0.1$. For the GP algorithm, the solid and dashed lines represent the result of the GP algorithm with $E_0 = E_g$. The light blue area illustrates the range of necessary measurement cost when we take $E_0 \in [E_g - 0.1, E_g + 0.1]$ in the GP algorithm.

When estimating the necessary measurement numbers, we need to numerically simulate the estimators $\hat{\mathbf{H}}$ and $\hat{\mathbf{S}}$, which can be regarded as the summations of the matrices $\mathbf{H}$ and $\mathbf{S}$ and random noise matrices $\hat{\mathbf{H}} - \mathbf{H}$ and $\hat{\mathbf{S}} - \mathbf{S}$, respectively. Considering Gaussian statistical error and the collective measurement protocol, the random matrices $\hat{\mathbf{H}} - \mathbf{H}$ and $\hat{\mathbf{S}} - \mathbf{S}$ can be numerically generated. The real and imaginary parts of the random matrix entries obey the Gaussian distribution with the mean value 0 (assume the estimation is unbiased for all algorithms) and the standard deviations are $C_{\mathbf{H}}/\sqrt{M}$ and $C_{\mathbf{S}}/\sqrt{M}$, respectively. Under collective measurement protocols, the random matrices have the same structure as $\mathbf{H}$ and $\mathbf{S}$.

L.2 Comparison using the thresholding method

Ref. [16] provides a detailed analysis of the thresholding procedure. The optimal thresholds are roughly close to the noise rate. Here, we set them to $10C_{\mathbf{S}}/\sqrt{M}$ to get relatively stable results. With the same instance $(\text{Heisenberg}, \text{chain}, d = 5)$, the results shown in Fig. 8 are similar to Fig. 7. This indicates that regardless of which method is used to deal with the ill-conditioned problem, GP algorithm outperforms other algorithms in terms of measurement costs.

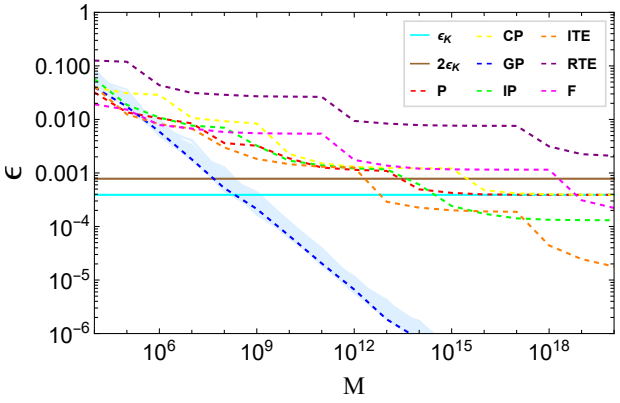
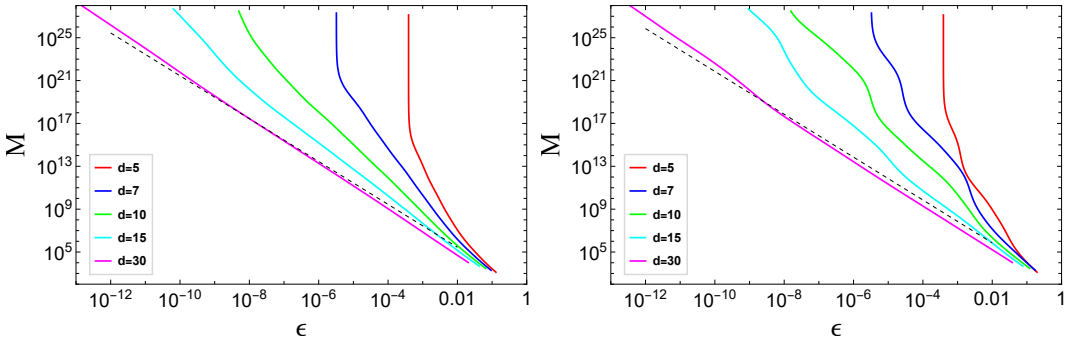
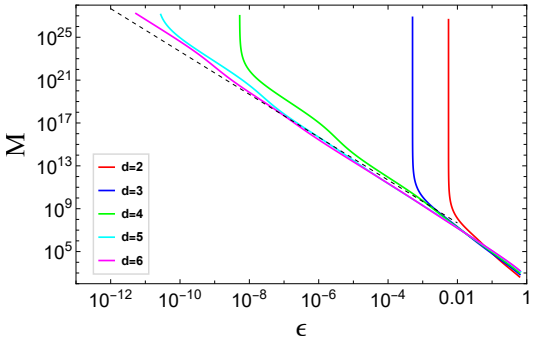


Figure 8: The absolute energy error ϵ and the corresponding necessary measurement numbers M when using the thresholding method. The instance is $(\text{Heisenberg}, \text{chain}, d = 5)$ and $E_g = -1$. Similar to Fig. 7, the dashed lines indicate the necessary measurement costs with $\kappa = 0.1$. The blue dashed line represents the result of the GP algorithm with $E_0 = E_g$. The light blue area illustrates the range of necessary measurement cost when we take $E_0 \in [E_g - 0.1, E_g + 0.1]$ in the GP algorithm.

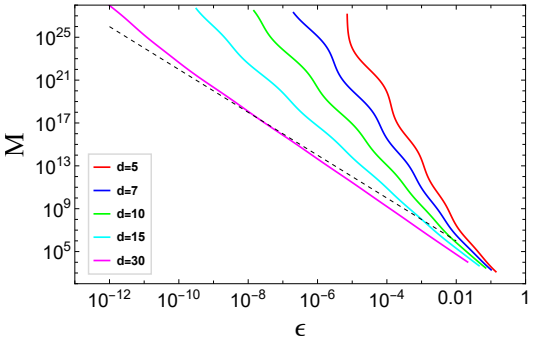


(a) Power

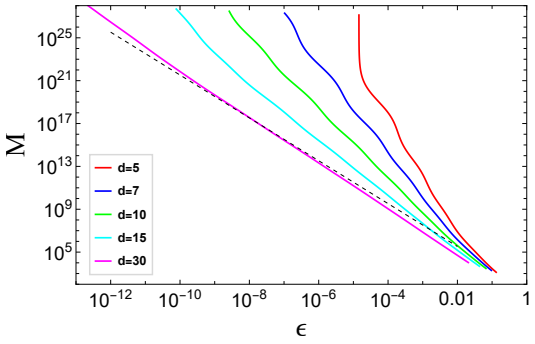
(b) Chebyshev polynomial



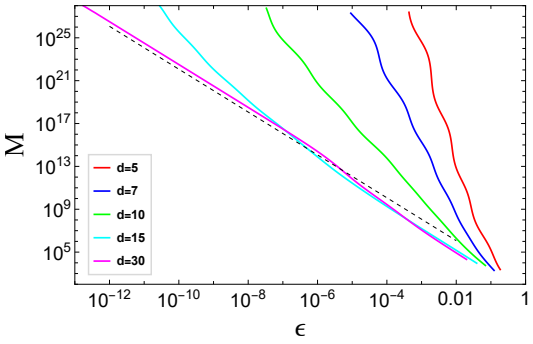
(c) Gaussian-power



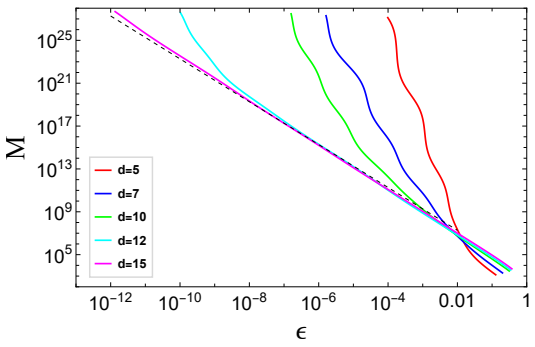
(d) Inverse power



(e) Imaginary-time evolution



(f) Real-time evolution



(g) Filter

Figure 9: The estimated measurement number M and the corresponding energy error ϵ at different d for quantum Krylov subspace diagonalisation algorithms. We take the 10-qubit Heisenberg chain model as an example. The dashed lines indicate the $1/\sqrt{M}$ scaling. The failure rate $\kappa = 0.1$. For the GP basis, E_0 is randomly chosen in $[E_g - 0.1, E_g + 0.1]$. For other bases, we use optimal parameters and minimum cost factors.

References

[1] Elbio Dagotto. "Correlated electrons in high-temperature superconductors". Reviews of Modern Physics **66**, 763 (1994).

[2] Michael R Wall and Daniel Neuhauser. "Extraction, through filter-diagonalization, of general quantum eigenvalues or classical normal mode frequencies from a small number of residues or a short-time segment of a signal. i. theory and application to a quantum-dynamics model". The Journal of chemical physics **102**, 8011–8022 (1995).

[3] Etienne Caurier, Gabriel Martínez-Pinedo, Fréderic Nowacki, Alfredo Poves, and AP Zuker. "The shell model as a unified view of nuclear structure". Reviews of modern Physics **77**, 427 (2005).

[4] Cornelius Lanczos. "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators". Journal of research of the National Bureau of Standards **45**, 255–282 (1950).

[5] Yousef Saad. "Numerical methods for large eigenvalue problems". Society for Industrial and Applied Mathematics. (2011).

[6] Åke Björck. "Numerical methods in matrix computations". Volume 59 of Texts in Applied Mathematics. Springer. (2015).

[7] Adolfo Avella and Ferdinando Mancini. "Strongly correlated systems". Volume 176 of Springer Series in Solid-State Sciences. Springer. (2013).

[8] Mario Motta, Chong Sun, Adrian T. K. Tan, Matthew J. O'Rourke, Erika Ye, Austin J. Minnich, Fernando G. S. L. Brandão, and Garnet Kin-Lic Chan. "Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution". Nature Physics **16**, 205–210 (2019).

[9] Kübra Yeter-Aydeniz, Raphael C Pooser, and George Siopsis. "Practical quantum computation of chemical and nuclear energy levels using quantum imaginary time evolution and lanczos algorithms". npj Quantum Information **6**, 1–8 (2020).

[10] Robert M. Parrish and Peter L. McMahon. "Quantum filter diagonalization: Quantum eigendecomposition without full quantum phase estimation" (2019). arXiv:1909.08925.

[11] Nicholas H. Stair, Renke Huang, and Francesco A. Evangelista. "A multireference quantum krylov algorithm for strongly correlated electrons". Journal of Chemical Theory and Computation **16**, 2236–2245 (2020).

[12] Tatiana A Bespalova and Oleksandr Kyrienko. "Hamiltonian operator approximation for energy measurement and ground-state preparation". PRX Quantum **2**, 030318 (2021).

[13] Jeffrey Cohn, Mario Motta, and Robert M Parrish. "Quantum filter diagonalization with compressed double-factorized hamiltonians". PRX Quantum **2**, 040352 (2021).

[14] Katherine Klymko, Carlos Mejuto-Zaera, Stephen J Cotton, Filip Wudarski, Miroslav Urbanek, Diptarka Hait, Martin Head-Gordon, K Birgitta Whaley, Jonathan Moussa, Nathan Wiebe, et al. "Real-time evolution for ultracompact hamiltonian eigenstates on quantum hardware". PRX Quantum **3**, 020323 (2022).

[15] Cristian L Cortes and Stephen K Gray. "Quantum krylov subspace algorithms for ground-and excited-state energy estimation". Physical Review A **105**, 022417 (2022).

[16] Ethan N Epperly, Lin Lin, and Yuji Nakatsukasa. "A theory of quantum subspace diagonalization". SIAM Journal on Matrix Analysis and Applications **43**, 1263–1290 (2022).

[17] Yizhi Shen, Katherine Klymko, James Sud, David B. Williams-Young, Wibe A. de Jong, and Norm M. Tubman. "Real-Time Krylov Theory for Quantum Computing Algorithms". Quantum **7**, 1066 (2023).

[18] Oleksandr Kyriienko. "Quantum inverse iteration algorithm for programmable quantum simulators". npj Quantum Information **6**, 1–8 (2020).

[19] Kazuhiro Seki and Seiji Yunoki. "Quantum power method by a superposition of time-evolved states". PRX Quantum **2**, 010333 (2021).

[20] William Kirby, Mario Motta, and Antonio Mezzacapo. "Exact and efficient Lanczos method on a quantum computer". Quantum **7**, 1018 (2023).

[21] Ryan Babbush, Craig Gidney, Dominic W. Berry, Nathan Wiebe, Jarrod McClean, Alexandru Paler, Austin Fowler, and Hartmut Neven. "Encoding electronic spectra in quantum circuits with linear t complexity". Phys. Rev. X **8**, 041015 (2018).

[22] Joonho Lee, Dominic W. Berry, Craig Gidney, William J. Huggins, Jarrod R. McClean, Nathan Wiebe, and Ryan Babbush. "Even more efficient quantum computations of chemistry through tensor hypercontraction". PRX Quantum **2**, 030305 (2021).

[23] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O'brien. "A variational eigenvalue solver on a photonic quantum processor". Nature communications **5**, 4213 (2014).

[24] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. "Barren plateaus in quantum neural network training landscapes". Nature communications **9**, 4812 (2018).

[25] Beresford N. Parlett. "The symmetric eigenvalue problem". Society for Industrial and Applied Mathematics. (1998).

[26] Robert M Parrish, Edward G Hohenstein, Peter L McMahon, and Todd J Martínez. "Quantum computation of electronic transitions using a variational quantum eigensolver". Physical review letters **122**, 230401 (2019).

[27] Ken M Nakanishi, Kosuke Mitarai, and Keisuke Fujii. "Subspace-search variational quantum eigensolver for excited states". Physical Review Research **1**, 033062 (2019).

[28] William J Huggins, Joonho Lee, Unpil Baek, Bryan O'Gorman, and K Birgitta Whaley. "A non-orthogonal variational quantum eigensolver". New Journal of Physics **22**, 073009 (2020).

[29] Nicholas H. Stair, Cristian L. Cortes, Robert M. Parrish, Jeffrey Cohn, and Mario Motta. "Stochastic quantum krylov protocol with double-factorized hamiltonians". Phys. Rev. A **107**, 032414 (2023).

[30] Jarrod R. McClean, Mollie E. Kimchi-Schwartz, Jonathan Carter, and Wibe A. de Jong. "Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states". Phys. Rev. A **95**, 042308 (2017).

[31] Jarrod R McClean, Zhang Jiang, Nicholas C Rubin, Ryan Babbush, and Hartmut Neven. "Decoding quantum errors with subspace expansions". Nature communications **11**, 636 (2020).

[32] Nobuyuki Yoshioka, Hideaki Hakoshima, Yuichiro Matsuzaki, Yuuki Tokunaga, Yasunari Suzuki, and Suguru Endo. "Generalized quantum subspace expansion". Phys. Rev. Lett. **129**, 020502 (2022).

[33] Artur K Ekert, Carolina Moura Alves, Daniel KL Oi, Michał Horodecki, Paweł Horodecki, and Leong Chuan Kwek. "Direct estimations of linear and nonlinear functionals of a quantum state". Physical review letters **88**, 217901 (2002).

[34] Mingxia Huo and Ying Li. "Error-resilient monte carlo quantum simulation of imaginary time". Quantum **7**, 916 (2023).

[35] Pei Zeng, Jinzhao Sun, and Xiao Yuan. "Universal quantum algorithmic cooling on a quantum computer" (2021). arXiv:2109.15304.

[36] Andrew M Childs and Nathan Wiebe. "Hamiltonian simulation using linear combinations of unitary operations". Quantum Information & Computation **12**, 901–924 (2012).

[37] Paul K. Faehrmann, Mark Steudtner, Richard Kueng, Maria Kieferova, and Jens Eisert. "Randomizing multi-product formulas for Hamiltonian simulation". Quantum **6**, 806 (2022).

[38] Thomas E O'Brien, Stefano Polla, Nicholas C Rubin, William J Huggins, Sam McArdle, Sergio Boixo, Jarrod R McClean, and Ryan Babbush. "Error mitigation via verified phase estimation". PRX Quantum **2**, 020317 (2021).

[39] Sirui Lu, Mari Carmen Bañuls, and J Ignacio Cirac. "Algorithms for quantum simulation at finite energies". PRX Quantum **2**, 020321 (2021).

[40] G.B. Arfken, H.J. Weber, and F.E. Harris. "Mathematical methods for physicists: A comprehensive guide". Elsevier. (2012).

[41] Seth Lloyd. "Universal quantum simulators". Science **273**, 1073–1078 (1996).

[42] Dominic W Berry, Graeme Ahokas, Richard Cleve, and Barry C Sanders. "Efficient quantum algorithms for simulating sparse hamiltonians". Communications in Mathematical Physics **270**, 359–371 (2007).

[43] Nathan Wiebe, Dominic Berry, Peter Høyer, and Barry C Sanders. "Higher order decompositions of ordered operator exponen-

tials". Journal of Physics A: Mathematical and Theoretical **43**, 065203 (2010).

[44] Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. "Simulating hamiltonian dynamics with a truncated taylor series". Physical review letters **114**, 090502 (2015).

[45] Richard Meister, Simon C. Benjamin, and Earl T. Campbell. "Tailoring Term Truncations for Electronic Structure Calculations Using a Linear Combination of Unitaries". Quantum **6**, 637 (2022).

[46] Earl Campbell. "Random compiler for fast hamiltonian simulation". Physical review letters **123**, 070503 (2019).

[47] Yongdan Yang, Bing-Nan Lu, and Ying Li. "Accelerated quantum monte carlo with mitigated error on noisy quantum computer". PRX Quantum **2**, 040361 (2021).

[48] Philip W Anderson. "The resonating valence bond state in la2cuo4 and superconductivity". science **235**, 1196–1198 (1987).

[49] Patrick A Lee, Naoto Nagaosa, and Xiao-Gang Wen. "Doping a mott insulator: Physics of high-temperature superconductivity". Reviews of modern physics **78**, 17 (2006).

[50] Kazuhiro Seki, Tomonori Shirakawa, and Seiji Yunoki. "Symmetry-adapted variational quantum eigensolver". Phys. Rev. A **101**, 052340 (2020).

[51] Roger A. Horn and Charles R. Johnson. "Matrix analysis". Cambridge University Press. (2012). 2 edition.

[52] Germund Dahlquist and Åke Björck. "Numerical methods in scientific computing, volume i". Society for Industrial and Applied Mathematics. (2008).

[53] William Kirby. "Analysis of quantum krylov algorithms with errors" (2024). arXiv:2401.01246.

[54] `https://github.com/ZongkangZhang/QKSD`.

[55] Wassily Hoeffding. "Probability inequalities for sums of bounded random variables". Journal of the American Statistical Association **58**, 13–30 (1963).

[56] Joel A. Tropp. "An introduction to matrix concentration inequalities". Foundations and Trends® in Machine Learning **8**, 1–230 (2015).

[57] Raymundo Albert and Cecilia G. Galarza. "Model order selection for sum of complex exponentials". In 2021 IEEE URUCON. Pages 561–565. (2021).

[58] D. Babusci, G. Dattoli, and M. Quattromini. "On integrals involving hermite polynomials". Applied Mathematics Letters **25**, 1157–1160 (2012).

[59] Alexei Yu Kitaev, Alexander Shen, and Mikhail N Vyalyi. "Classical and quantum computation". Graduate studies in mathematics. American Mathematical Society. (2002).

[60] Julia Kempe, Alexei Kitaev, and Oded Regev. "The complexity of the local hamiltonian problem". Siam journal on computing **35**, 1070–1097 (2006).

[61] Chris Cade, Marten Folkertsma, Sevag Gharibian, Ryu Hayakawa, François Le Gall, Tomoyuki Morimae, and Jordi Weggemans. "Improved Hardness Results for the Guided Local Hamiltonian Problem". In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, 50th International Colloquium on Automata, Languages, and Programming (ICALP 2023). Volume 261 of Leibniz International Proceedings in Informatics (LIPIcs), pages 32:1–32:19. Dagstuhl, Germany (2023). Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[62] Seunghoon Lee, Joonho Lee, Huanchen Zhai, Yu Tong, Alexander M Dalzell, Ashutosh Kumar, Phillip Helms, Johnnie Gray, Zhi-Hao Cui, Wenyuan Liu, et al. "Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry". Nature communications **14**, 1952 (2023).