Quantum phase processing and its applications in estimating phase and entropies

Youle Wang,[1,2] Lei Zhang,[1,3] Zhan Yu,[1,4] and Xin Wang[1,3,*]

[1]*Institute for Quantum Computing, Baidu Research, Beijing 100193, China*
[2]*School of Software, Nanjing University of Information Science and Technology, Nanjing 210044, China*
[3]*Thrust of Artificial Intelligence, Information Hub, Hong Kong University of Science and Technology (Guangzhou), Guangdong 511455, China*
[4]*Centre for Quantum Technologies, National University of Singapore, 117543 Singapore*

(Received 13 July 2023; accepted 3 November 2023; published 13 December 2023)

Quantum computing can provide speedups in solving many problems as the evolution of a quantum system is described by a unitary operator in an exponentially large Hilbert space. Such unitary operators change the phase of their eigenstates and make quantum algorithms fundamentally different from their classical counterparts. Based on this unique principle of quantum computing, we develop an algorithmic toolbox, quantum phase processing, that can directly apply arbitrary trigonometric transformations to eigenphases of a unitary operator. The quantum phase processing circuit is constructed simply, consisting of single-qubit rotations and controlled unitaries, typically using only one ancilla qubit. Besides the capability of phase transformation, quantum phase processing in particular can extract the eigeninformation of quantum systems by simply measuring the ancilla qubit, making it naturally compatible with indirect measurement. Quantum phase processing complements another powerful framework known as quantum singular value transformation and leads to more intuitive and efficient quantum algorithms for solving problems that are particularly phase related. As a notable application, we propose a quantum phase estimation algorithm without quantum Fourier transform, which requires the fewest ancilla qubits and matches the best performance so far. We further exploit the power of our method by investigating a plethora of applications in Hamiltonian simulation, entanglement spectroscopy, and quantum entropy estimation, demonstrating improvements or optimality for almost all cases.

DOI: 10.1103/PhysRevA.108.062413

I. INTRODUCTION

Quantum computers provide a computational framework that can solve certain problems dramatically faster than classical machines. Quantum computing has been applied in many important tasks, including breaking encryption [1], searching databases [2], and simulating quantum evolution [3]. Recent advances in quantum computing show that quantum singular value transformation (QSVT), introduced by Gilyén *et al.* [4], has led to a unified framework of the most known quantum algorithms [5], including amplitude amplification [4], quantum walks [4], phase estimation [5,6], and Hamiltonian simulations [7–10]. This framework can further be used to develop new quantum algorithms such as quantum entropy estimation [11–13], fidelity estimation [14], ground-state preparation, and ground-energy estimation [15–17].

The framework of QSVT originated from a technique called quantum signal processing (QSP) [18,19]. By interleaving single-qubit signal unitaries and signal processing unitaries, QSP is able to implement a transformation of the signal in SU(2). There are several conventions of QSP varied by choosing different signal unitaries. In the construction of QSVT, Gilyén *et al.* [4] chose the signal unitary to be a reflection and then extended the signal unitary to a multiqubit block encoding with the idea of qubitization [7], which naturally leads to a polynomial transformation of the singular values of a block-encoded linear operator. The achievable polynomial transformations in QSVT are determined by reflection-based QSP, which has parity constraints or limitations, i.e., it can implement either an even polynomial or an odd one. Thus, to achieve a general transformation in QSVT, one might have to apply techniques such as linear combination of unitaries [20] and amplitude amplification [21,22], which take extra resources such as ancilla qubits. Based on a convention of QSP using z rotation as the signal unitary [23,24], Yu *et al.* [25] developed an improved version of QSP that overcame the parity limitation by adding an extra signal processing unitary, which could implement arbitrary complex trigonometric polynomials on one-qubit quantum systems and also provides insight into the understanding quantum neural networks.

For the signal unitary being a z rotation, the corresponding trigonometric QSP naturally possesses the ability of processing phases, which indeed plays a central role in many quantum algorithms. For example, the trick of phase kickback, where the phase of the target qubits is kicked back to the ancilla qubit, is intensively used almost everywhere in quantum computing. With the aid of controlled-unitary (controlled-U) gates, many quantum algorithms utilize phase kickback to extract information of large unitary operations from phases of ancilla qubits, including the quantum phase estimation [26,27], the SWAP test [28,29], the Hadamard test [30], and the one-clean-qubit model [31]. Hence, it is of great interest and necessity to explore a generalized formalism that could

[*] felixxinwang@hkust-gz.edu.cn

interpret those phase-related quantum algorithms, which may further lead to new quantum algorithms and help us better exploit the power of quantum signal processing. Consequently, it is natural to investigate and develop a multiqubit extension of the improved trigonometric QSP in Ref. [25].

In this work, we generalize the trigonometric QSP and propose an algorithmic toolbox called quantum phase processing (QPP). This toolbox has the ability to apply arbitrary trigonometric transformations to eigenphases of a unitary operator. Besides achieving the eigenphase transformation, QPP is also natively compatible with indirect measurements, enabling it to extract the eigeninformation of quantum systems by measuring a single ancilla qubit. We further employ this toolbox to design efficient quantum algorithms for solving various problems. First, we use the idea of binary search to develop an efficient phase estimation algorithm without using the quantum Fourier transform, requiring only one ancilla qubit. Such an algorithm can be applied to solve factoring problems and amplitude estimations. Second, we show that QPP can be applied to simulate the time evolution under a Hamiltonian H with access to a block encoding of H. This method is in the same spirit as QSP-based Hamiltonian simulation [7,19], which also matches the optimal query complexity. Third, we propose a generic method to estimate quantum entropies, including von Neumann entropy, quantum relative entropy, and the family of quantum Rényi entropies [32]. Despite the fact that QPP could be combined with amplitude estimation to achieve a quadratic speedup, we present algorithms that repeatedly measure the single ancilla qubit to estimate entropies rather than using amplitude estimation, demonstrating its compatibility with indirect measurements. Overall, QPP provides a powerful algorithmic toolbox to exploit quantum applications and delivers a different perspective on understanding and designing quantum algorithms.

The structure of this paper is presented as follows. Section II introduces the structure and principal capability of quantum phase processing. In Sec. III we propose the quantum phase search algorithm, analyze the performance of the algorithm, and make a brief comparison with previous works. Section V interprets the method of Hamiltonian simulation in the QPP structure. In Sec. IV we develop a generic approach for quantum entropy estimation and further showcase the methods of estimating von Neumann entropies, quantum relative entropies, and quantum Rényi entropies; then we compare our algorithms with prior methods. Proofs and further discussion of this work are given in the Appendixes.

II. QUANTUM PHASE PROCESSING

A. Quantum signal processing

We first review the concept of QSP. Quantum signal processing was introduced by Low *et al.* [18], who showed how to transform a 2 × 2 signal unitary $R_x(x) = e^{ix\sigma_x}$ into a target unitary whose entries are some transformations of the signal x. The approach is to apply the signal unitary $R_x(x)$ interleaved with some signal processing unitaries $R_z(\phi)$, i.e.,

$$R_z(\phi_0)R_x(x)R_z(\phi_1)R_x(x)\cdots R_x(x)R_z(\phi_k). \tag{1}$$

Gilyén *et al.* [4] modified the signal unitary as a reflection and explicitly showed that the transformation corresponds to a Chebyshev polynomial of the signal x. Another common convention of QSP is to choose the signal unitary to be a z rotation $R_z(x)$ with signal processing unitaries being x rotations $R_x(\phi)$ [23,24], which corresponds to a trigonometric polynomial of the signal x. Different types of QSP and their relationships were summarized by Martyn *et al.* [5]. Observe that both of these two conventions of QSP have constraints on the achievable polynomials: For the Chebyshev QSP, each entry is a polynomial with either even or odd parity; for the trigonometric QSP, each entry is a trigonometric polynomial (in the exponential form) with either real or imaginary coefficients. As a result, the technique of linear combination of unitaries [20] might be required for these conventions of QSP to implement a general polynomial transformation, which requires extra ancilla qubits. In a recent work, Yu *et al.* [25] overcame the constraints by adding an extra signal processing unitary in each iteration so that one could implement arbitrary complex trigonometric polynomial transformation in a single QSP. Our work is heavily based on this improved trigonometric QSP, which is defined as

$$W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x) := R_z(\omega)R_y(\theta_0)R_z(\phi_0)\prod_{l=1}^{L}R_z(x)R_y(\theta_l)R_z(\phi_l), \tag{2}$$

where $L \in \mathbb{N}$ is the number of layers, and $\omega \in \mathbb{R}$, $\boldsymbol{\theta} = (\theta_0, \theta_1, \ldots, \theta_L) \in \mathbb{R}^{L+1}$, and $\boldsymbol{\phi} = (\phi_0, \phi_1, \ldots, \phi_L) \in \mathbb{R}^{L+1}$ are parameters. The quantum circuits of different QSP conventions and their realizable polynomials are presented in Table IV.

The following lemma characterizes the correspondence between trigonometric QSP and complex trigonometric polynomials. The initial version of Lemma 1 first introduced in [25] is in the form of quantum neural networks. Here we restate the lemma in the formalism of QSP without changing the results.

Lemma 1 (trigonometric quantum signal processing). There exist $\omega \in \mathbb{R}$, $\boldsymbol{\theta} = (\theta_0, \theta_1, \ldots, \theta_L) \in \mathbb{R}^{L+1}$, and $\boldsymbol{\phi} = (\phi_0, \phi_1, \ldots, \phi_L) \in \mathbb{R}^{L+1}$ such that

$$W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x) = \begin{bmatrix} P(x) & -Q(x) \\ Q^*(x) & P^*(x) \end{bmatrix} \tag{3}$$

if and only if Laurent polynomials $P, Q \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ satisfy the following conditions: (a) $\deg(P) \leqslant L$ and $\deg(Q) \leqslant L$, (b) P and Q have parity[1] $L \bmod 2$, and (c) $|P(x)|^2 + |Q(x)|^2 = 1\,\forall\, x \in \mathbb{R}$.

Lemma 1 demonstrates a decomposition of QSP $W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)$ into complex Laurent polynomials, as well as a construction of QSP from complex Laurent polynomials. From condition (b) of Lemma 1, it seems that P and Q still have parity constraints, i.e., they have either parity 0 or 1, but in fact Laurent polynomials in $\mathbb{C}[e^{ix/2}, e^{-ix/2}]$ with parity 0 are essentially complex trigonometric polynomials in $\mathbb{C}[e^{ix}, e^{-ix}]$ without parity constraints. The proof of this theorem also provides an algorithm that calculates angles ω, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$ in $O(\text{poly}(L))$

[1] For a Laurent polynomial $P \in \mathbb{C}[z, z^{-1}]$, P has parity 0 if all coefficients corresponding to odd powers of z are 0, and similarly P has parity 1 if all coefficients corresponding to even powers of z are 0.

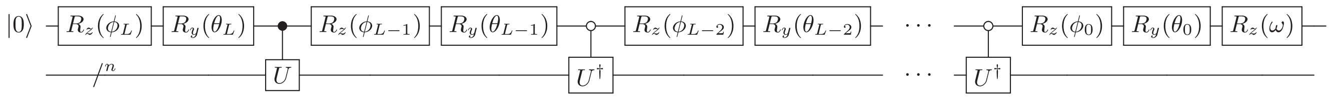


FIG. 1. General circuit for quantum phase processing $V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)$; here the number of layers L is an even integer.

operations; one can refer to Algorithm 3 in Appendix B 1 for more details. There are also many other methods to compute the angles; see, e.g., Refs. [23,24,33,34]. It can be inferred from Lemma 1 that if $P(x)$ satisfies the parity constraint and $|P(x)| \leqslant 1$ for all $x \in \mathbb{R}$, then there exists a corresponding $Q(x)$ satisfying the three conditions. A detailed analysis can be found in Appendix B 1.

Following the trigonometric QSP construction and decomposition in Ref. [25], we are interested in how to represent the trigonometric polynomial transformation. One way is to project out $P(x)$ from $W_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}$, i.e., the $\langle 0|W_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}|0\rangle$.

Corollary 1. For any complex-valued trigonometric polynomial $F(x) = \sum_{j=-L}^{L} c_j e^{ijx}$ with $|F(x)| \leqslant 1$ for all $x \in \mathbb{R}$, there exist $\omega \in \mathbb{R}$ and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{2L+1}$ such that for all $x \in \mathbb{R}$,

$$\langle 0|W^{2L}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)|0\rangle = F(x). \tag{4}$$

Moreover, based on the fact that any non-negative real-valued trigonometric polynomial can be decomposed as a product between a Laurent polynomial in $\mathbb{C}[e^{ix/2}, e^{-ix/2}]$ and its complex conjugate, the trigonometric polynomial can be represented by the expectation value of measuring a Pauli-Z observable with respect to the state $W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}|0\rangle$.

Corollary 2. For any real-valued trigonometric polynomial $F(x) = \sum_{j=-L}^{L} c_j e^{ijx}$ with $|F(x)| \leqslant 1$ for all $x \in \mathbb{R}$, there exist $\omega \in \mathbb{R}$ and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{L+1}$ such that for all $x \in \mathbb{R}$,

$$f_W(x) := \langle 0|W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)^\dagger Z W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)|0\rangle = F(x). \tag{5}$$

The key of this corollary is that, for any real-valued trigonometric polynomial $F(x)$ with degree L and satisfying $|F(x)| \leqslant 1$, we could always find a complex root $P \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ such that $PP^* = [1 + F(x)]/2$, as proved in Appendix B 1, and then the expectation value of measuring Z observable turns out to be $F(x)$. However, such a root decomposition does not hold in the polynomial space $\mathbb{C}[x]$. For instance, letting $F(x) = x$, there does not exist $P \in \mathbb{C}[x]$ such that $PP^* = [1 + F(x)]/2$. Thus the Chebyshev QSP [4,18,19] could not directly use the Pauli-Z measurement to obtain a Chebyshev polynomial.

As a summary of this section, the construction and functions of the trigonometric QSP are briefly reviewed. In particular, the trigonometric QSP is capable of implementing arbitrary complex trigonometric polynomials by projection or measuring the Pauli-Z observable in a single-qubit system. These properties are the fundamental reasons why its generalized version can achieve improvements in aspects of phase estimation and entropy estimation, as will be demonstrated in later sections.

B. Processing phases of high-dimensional unitaries

Although the model of QSP provides a systematic method to make arbitrary polynomial transformations, it only works on a qubitlike quantum system. Gilyén *et al.* [4] proposed a multiqubit lifted version of the Chebyshev QSP, called quantum singular value transformation, which could efficiently apply Chebyshev polynomial transformations to the singular values of a linear operator embedded in a larger unitary. In this work, we consider a similar extension of the trigonometric QSP and establish a quantum phase processing algorithmic toolbox that could apply arbitrary trigonometric transformations to eigenphases of a unitary matrix U. The structure of QPP generalizes the trigonometric QSP by replacing the input signal x with the phases of a higher-dimensional unitary matrix. For an even $L \in \mathbb{N}$ and angle parameters $\omega \in \mathbb{R}$ and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{L+1}$, we define the quantum phase processor of the n-qubit unitary U as

$$V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U) := R_z^{(0)}(\omega) R_y^{(0)}(\theta_0) R_z^{(0)}(\phi_0) \left(\prod_{l=1}^{L/2} \begin{bmatrix} U^\dagger & 0 \\ 0 & I^{\otimes n} \end{bmatrix} R_y^{(0)}(\theta_{2l-1}) R_z^{(0)}(\phi_{2l-1}) \begin{bmatrix} I^{\otimes n} & 0 \\ 0 & U \end{bmatrix} R_y^{(0)}(\theta_{2l}) R_z^{(0)}(\phi_{2l}) \right), \tag{6}$$

where $R_y^{(0)}$ and $R_z^{(0)}$ are rotation gates applied on the first qubit. For an odd $L \in \mathbb{N}$, we apply an extra layer

$$\begin{bmatrix} U^\dagger & 0 \\ 0 & I^{\otimes n} \end{bmatrix} R_y^{(0)}(\theta_L) R_z^{(0)}(\phi_L)$$

to $V^{L-1}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)$. The quantum circuit of $V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)$ is shown in Fig. 1.

One could find that QPP simply replaces the signal unitary $R_z(x)$ in QSP with interleaved controlled U and controlled $U^\dagger$. We note that such a construction of using controlled unitaries was first purposed by Low and Chuang [19], namely, the signal transduction, which was frequently used in various works [7,17,33]. The intuition lying behind the extension is that the controlled U and its inverse are naturally multiqubit analogs of R_z gates. To better understand how rotation gates in the first ancilla qubit process the phase of the target unitary U, we analyze the eigenspace decomposition of QPP.

Lemma 2 (eigenspace decomposition of QPP). Suppose U is an n-qubit unitary with spectral decomposition

$$U = \sum_{j=0}^{2^n-1} e^{i\tau_j} |\chi_j\rangle\langle\chi_j|. \tag{7}$$

For all $L \in \mathbb{N}$, $\omega \in \mathbb{R}$, and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{L+1}$ we have

$$V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U) = \bigoplus_{j=0}^{2^n-1} (e^{-i\tau_j/2})^{L \bmod 2} W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(\tau_j)_{\mathbb{B}_j}, \tag{8}$$

where $\mathbb{B}_j := \{|0, \chi_j\rangle, |1, \chi_j\rangle\}$.

The proof of this lemma is deferred to Appendix B 3. Lemma 2 shows that the eigenspace of QPP $V_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)$ coincides with that of the unitary U. Using this property, we generalize the single-qubit trigonometric QSP to the multiqubit QPP that could perform a trigonometric polynomial transformation on the eigenphase of the unitary U. In a spirit similar to previous works [4,23,24], we could measure the first ancilla qubit and achieve evolution of the input state upon postselection of the measurement result being $|0\rangle$, as shown in the following theorem.

Theorem 1 (quantum phase evolution). Given an n-qubit unitary $U = \sum_{j=0}^{2^n-1} e^{i\tau_j} |\chi_j\rangle\langle\chi_j|$, for any trigonometric polynomial $F(x) = \sum_{j=-L}^{L} c_j e^{ijx}$, with $|F(x)| \leqslant 1$ for all $x \in \mathbb{R}$, there exist $\omega \in \mathbb{R}$ and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{2L+1}$ such that

$$V^{2L}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U) = \begin{bmatrix} F(U) & \cdots \\ \cdots & \cdots \end{bmatrix}, \tag{9}$$

where $F(U) := \sum_{j=0}^{2^n-1} F(\tau_j)|\chi_j\rangle\langle\chi_j|$.

The proof of Theorem 1 trivially follows from combining Lemma 2 with Corollary 1, which we defer to Appendix B 4. The other way is to evaluate the trigonometric polynomial on the eigenphases and represent the result by the expectation value of measuring an observable on the first qubit.

Theorem 2 (quantum phase evaluation). Given an n-qubit unitary $U = \sum_{j=0}^{2^n-1} e^{i\tau_j}|\chi_j\rangle\langle\chi_j|$ and an n-qubit quantum state ρ, for any real-valued trigonometric polynomial $F(x) = \sum_{j=-L}^{L} c_j e^{ijx}$, with $|F(x)| \leqslant 1$ for all $x \in \mathbb{R}$, there exist $\omega \in \mathbb{R}$ and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{L+1}$ such that $\hat{\rho} = V_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)(|0\rangle\langle 0| \otimes \rho)V_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)^\dagger$ satisfies

$$f_V(U) := \operatorname{tr}[(Z^{(0)} \otimes I^{\otimes n})\hat{\rho}] = \sum_{j=0}^{2^n-1} p_j F(\tau_j), \tag{10}$$

where $p_j = \langle\chi_j|\rho|\chi_j\rangle$ and $Z^{(0)}$ is a Pauli-Z observable acting on the first qubit.

Theorem 2 is proved by combining Lemma 2 and Corollary 2, as shown in Appendix B 4. Theorem 2 shows that QPP is natively compatible with indirect measurements, which could represent the target trigonometric polynomial by probabilities of measuring the ancilla qubit. Such a property does not emerge in QSVT, since the Chebyshev QSP typically implements the target polynomial transformation using projection rather than the Pauli-Z measurement. Chebyshev's inequality dictates that an estimate of the expectation value within an additive error δ can be obtained by measuring the ancilla qubit $O(1/\delta^2)$ times. Alternatively, one could apply amplitude estimation [21] to estimate the value by calling the QPP circuit $O(1/\delta)$ times, which is quadratically more efficiently than classical sampling but with a larger circuit depth. In particular, since the ancilla qubit in QPP naturally works as a flag qubit, we could directly apply the iterative amplitude estimation [35] on QPP without using extra qubits.

Theorems 1 and 2 are vital elements of the QPP algorithmic toolbox, together demonstrating that QPP is a versatile and flexible toolbox for phase-related problems. First, QPP could act in a similar manner to QSVT but transforming eigenphases of a unitary rather than singular values of an embedded linear operator. Moreover, using block encoding and qubitization [7] enables QPP to process eigenvalues of an embedded operator as well, as presented in later applications. Second, QPP could extract the eigeninformation after desired transformations by simply measuring the single ancilla qubit, generalizing many indirect measurement methods like the Hadamard test, which could not be natively achieved by QSVT.

For the sake of notational simplicity, we omit the parameters L, ω, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$, writing QSP as $W(x)$ and QPP as $V(U)$ in the rest of this paper. The capabilities of QPP are summarized in Fig. 2. Next we will show that the structure of QPP is a powerful tool for designing efficient and intuitive quantum algorithms for solving various problems, including quantum phase estimation, Hamiltonian simulation, and quantum entropy estimation.

III. QUANTUM PHASE ESTIMATION

Quantum phase estimation is one of the most important and useful subroutines in quantum computing. The problem of phase estimation is formally defined as follows: Given a unitary U and an eigenstate $|\chi\rangle$ of U with eigenvalue $e^{i\tau}$, estimate the eigenphase τ up to an additive error δ. In this section, we will develop an efficient algorithm for quantum phase estimation based on QPP. Before proceeding, let us prepare by becoming familiar with the QPP toolbox. We start by considering a simple method of Hadamard test.

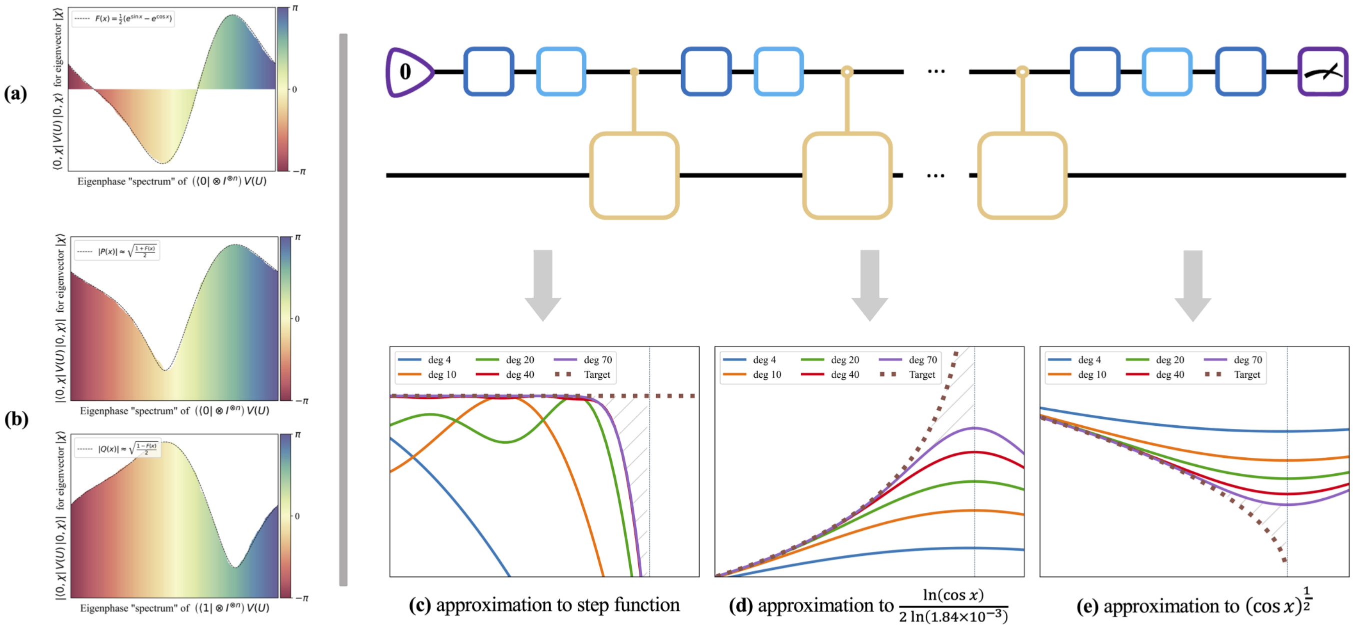
A. Example: Generalized Hadamard test

A common method to estimate the phase of a unitary is the Hadamard test, which solves the following problem: Given a unitary U and a state $|\chi\rangle$, estimate $\langle\chi|U|\chi\rangle$. The Hadamard test uses the measurement result of the ancillary qubit as a random variable whose expected value is the real part $\operatorname{Re}(\langle\chi|U|\chi\rangle)$. It can also estimate the imaginary part $\operatorname{Im}(\langle\chi|U|\chi\rangle)$ by adding a phase gate. We now show that QPP is a generalization of the Hadamard test.

Given a unitary U and a quantum state $|\chi\rangle$ such that $U|\chi\rangle = e^{i\tau}|\chi\rangle$, let $F(x) = \cos(x)$ be the target trigonometric polynomial for QPP. By Theorem 2 there exists a single-layer QPP $V(U)$ such that

$$\begin{aligned} f_V(U) &= \operatorname{tr}[\langle 0, \chi|V(U)^\dagger(Z^{(0)} \otimes I^{\otimes n})V(U)|0, \chi\rangle] \\ &= \cos(\tau) = \operatorname{Re}(\langle\chi|U|\chi\rangle). \end{aligned} \tag{11}$$

Specifically, by computing the angles ω, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$, one can find that the two rotation gates in the first ancilla qubit are essentially Hadamard gates, which means that QPP implements the Hadamard test. Similarly, letting the target trigonometric polynomial be $F(x) = \sin(x)$, then QPP can estimate $\sin(\tau) = \operatorname{Im}(\langle\chi|U|\chi\rangle)$. The phase τ could be obtained from $\cos(\tau)$ and $\sin(\tau)$. Furthermore, one can select trigonometric polynomials other than $\sin(x)$ and $\cos(x)$, which



(c) approximation to step function **(d)** approximation to $\frac{\ln(\cos x)}{2\ln(1.84\times10^{-3})}$ **(e)** approximation to $(\cos x)^{\frac{1}{2}}$

FIG. 2. Illustration of the capabilities of quantum phase processing. A trigonometric transformation $F(x) = \frac{1}{2}(e^{\sin x} - e^{\cos x})$ of eigenphases of an n-qubit unitary U can be represented by a QPP circuit $V(U)$ in two different ways. (a) One approach is to directly simulate the target function. By postselecting the ancilla qubit to be $|0\rangle$, the function transformations of phases are encoded into the amplitudes of eigenvectors. (b) The other approach is to retrieve the target function by the expectation value of measuring a Pauli-Z observable on the ancilla qubit. More intuitively, if the input state of circuit $V(U)$ is a maximally mixed state, then the sum of eigenphase transformations is approximately the square difference between two shaded areas in (b) multiplied by a constant. (c)–(e) Three examples for the QPP circuits to approximate functions near a discontinuity, which will be useful in applications of quantum phase estimation and quantum entropy estimation. Here the quantity "deg" refers to the degree of trigonometric polynomials, where every increase of such a degree takes an extra layer in a QPP circuit.

yields a generalization of the Hadamard test. For example, QPP with a trigonometric polynomial $F(x)$ that approximates the function $f(x) = x/2\pi$ could directly estimate the phase τ.

Although QPP can implement a generalized Hadamard test to estimate the phase, the input state $|\chi\rangle$ is required to be an eigenstate of the target unitary. However, quite often we are given a superposition of eigenstates instead of a pure eigenstate, such as in the factoring problem. In addition, measuring the ancilla qubit $O(1/\delta^2)$ times is necessary to estimate the expected value with an additive error δ, despite the fact that the complexity can be improved via amplitude estimation. Can we do better? As demonstrated in the following section, we can further employ QPP to construct a more efficient phase estimation algorithm that accepts a superposition of eigenstates, without using amplitude estimation.

B. Quantum phase searching

As introduced in Sec. II, QPP can directly process the eigenphases of the target unitary, which allows us to classify the eigenphases. The main idea is to use a trigonometric polynomial to approximate a step function so that we could utilize QPP to locate the eigenphases by a binary search procedure. We first show that QPP can classify the eigenphases of U.

Lemma 3 (phase classification). Given a unitary $U = \sum_{j=0}^{2^n-1} e^{i\tau_j}|\chi_j\rangle\langle\chi_j|$, then for any $\Delta \in (0, \pi)$ and $\varepsilon \in (0, 1)$, there exists a QPP circuit $V(U)$ of $L = O(\frac{1}{\Delta}\ln\frac{1}{\varepsilon})$ layers such that

$$V(U)|0,\chi_k\rangle = \begin{cases} \sqrt{1-\varepsilon_k}|0,\chi_k\rangle + \sqrt{\varepsilon_k}|1,\chi_k\rangle & \text{if } \tau_k \in [\Delta, \pi - \Delta) \\ \sqrt{\varepsilon_k}|0,\chi_k\rangle + \sqrt{1-\varepsilon_k}|1,\chi_k\rangle & \text{if } \tau_k \in (-\pi + \Delta, -\Delta] \end{cases} \tag{12}$$

for $0 \leqslant k < 2^n$, where $\varepsilon_k \in (0, \varepsilon)$.

The proof of Lemma 3 is deferred to Appendix C 1. By phase kickback, measuring the ancilla qubit determines which subinterval the eigenphase τ belongs to with probability at least $1 - \varepsilon$. Next we apply a phase shift $e^{i\zeta}$ to U to move to the middle point of the designated subinterval so that $V(e^{i\zeta}U)$ determines the next subinterval. Using this fascinating property, repeating the binary search procedure shrinks the phase interval until QPP cannot determine next subintervals, i.e., $\tau \in [\zeta_l, \zeta_r]$ and $|\zeta_r - \zeta_l| \approx 2\Delta$. See the phase interval search procedure in Algorithm 1 for details.

Algorithm 1. Phase interval search.

```
Require: A unitary $U$, an eigenstate $|\chi\rangle$ of $U$ with eigenvalue $e^{i\tau}$,
    an interval $[\zeta_l, \zeta_r]$, a $\Delta \in (0, \frac{1}{2})$, an $\varepsilon \in (0, 1)$, and an integer
    $\mathcal{Q}$.
Ensure: An updated interval $[\zeta_l, \zeta_r]$ such that $\tau \in [\zeta_l, \zeta_r]$ and
    $\zeta_r - \zeta_l = 2(\Delta + \frac{\pi}{2^{\mathcal{Q}+1}})$.
for $j = 0, \ldots, \mathcal{Q} - 1$ do
    Set the middle point $\zeta_m = \frac{\zeta_l + \zeta_r}{2}$.
    Construct QPP circuit $V_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(e^{-i\zeta_m}U)$ in Lemma 3 according
    to $\Delta$ and $\varepsilon$.
    Apply the circuit to the state $|0, \chi\rangle$ and measure the ancilla
    qubit. If $\zeta_r - \zeta_l > 2\pi - 2\Delta$, update
        $[\zeta_l, \zeta_r] \leftarrow \begin{cases} [\zeta_m - \Delta, \zeta_r + \Delta] & \text{if the outcome is 0} \\ [\zeta_l - \Delta, \zeta_m + \Delta] & \text{if the outcome is 1;} \end{cases}$
    otherwise
        $[\zeta_l, \zeta_r] \leftarrow \begin{cases} [\zeta_m - \Delta, \zeta_r], & \text{if the outcome is 0} \\ [\zeta_l, \zeta_m + \Delta], & \text{if the outcome is 1.} \end{cases}$
end for
Output the interval $[\zeta_l, \zeta_r]$.
```

Algorithm 2. Quantum phase search.

```
Require: A unitary $U$, an eigenstate $|\chi\rangle$ of $U$ with eigenvalue $e^{i\tau}$,
    a $\Delta \in (0, \frac{1}{2})$, an $\varepsilon \in (0, 1)$, and a $\delta \in (0, 1)$.
Ensure: A phase $\bar{\tau} \in \mathbb{R}$ such that $|\bar{\tau} - \tau| < \delta$.
Set $\mathcal{Q} \leftarrow \lceil \ln \frac{2\pi}{1-2\Delta} \rceil$, $\bar{\Delta} \leftarrow \Delta + \frac{\pi}{2^{\mathcal{Q}+1}}$, $\mathcal{T} \leftarrow \lceil \frac{\ln \delta}{\ln \Delta} \rceil$, and
    $d \leftarrow \lfloor 1/\bar{\Delta} \rfloor$. Initialize the interval $[\zeta_l, \zeta_r] \leftarrow [-\pi, \pi]$.
for $t = 0, \ldots, \mathcal{T} - 1$ do
    Update the interval by the phase interval search (PIS)
    procedure
        $[\zeta_l, \zeta_r] \leftarrow \mathrm{PIS}\left(U, |\chi\rangle, [\zeta_l, \zeta_r], \Delta, \frac{\varepsilon}{\mathcal{Q}\mathcal{T}}, \mathcal{Q}\right)$.
    Store the middle point of interval $\zeta_m^{(t)} \leftarrow \frac{\zeta_l + \zeta_r}{2}$.
    Update $U \leftarrow [e^{-i\zeta_m^{(t)}}U]^d$ and
        $[\zeta_l, \zeta_r] \leftarrow [d(\zeta_l - \zeta_m^{(t)}), d(\zeta_r - \zeta_m^{(t)})]$.
end for
Output $\bar{\tau} \leftarrow \sum_{t=0}^{\mathcal{T}-1} \zeta_m^{(t)} d^{-t}$.
```

The phase interval search procedure will shrink the phase interval to a length $2(\Delta + \frac{\pi}{2^{\mathcal{Q}+1}})$ with probability at least $(1 - \varepsilon)^{\mathcal{Q}}$, where $\mathcal{Q}$ is the number of repetitions. Now the phase interval is too narrow for phase classification. We apply QPP on $(e^{i\zeta}U)^d$ for some appropriate integer d so that the binary search procedure can continue to locate the amplified phase $d\tau \in [d\zeta_l, d\zeta_r]$. Repeating the entire procedure gives an estimation of phase τ up to required precision δ, as shown in Algorithm 2.

Note that Algorithm 2 could accept a superposition of eigenstates as the input, since eigenstates whose eigenvalues disagree with measurement results of the ancilla qubit will be filtered out, and finally the state in the main register converges to a single eigenstate of U at the end of the quantum phase search algorithm. We conclude the discussion in Theorem 3.

Theorem 3 (complexity of quantum phase search). Given an n-qubit unitary U and an eigenstate $|\chi\rangle$ of U with eigenvalue $e^{i\tau}$, Algorithm 2 can use one ancilla qubit and $O(\frac{1}{\delta}\ln(\frac{1}{\varepsilon}\ln\frac{1}{\delta}))$ queries to the controlled U and its inverse to obtain an estimation of τ up to δ precision with probability at least $1 - \varepsilon$.

We can see that the quantum phase search algorithm provides a nearly quadratic speedup compared to the Hadamard test method in Sec. III A. More importantly, the phase search algorithm does not require the input state being an eigenstate of U, making it more versatile for solving specific problems like the factoring problem.

C. Comparison to related works

The quantum phase estimation method originally purposed to solve the Abelian stabilizer problem [26,36] was found to work for general unitaries. The most well-known version of the phase estimation method [26] queries the controlled U, $O(\frac{1}{\delta})$ times and applies the inverse quantum Fourier transform (QFT) [37] to estimate the eigenphase of a unitary U with precision δ and success probability at least $4/\pi^2$. The success probability can be boosted to $1 - \varepsilon$ by using additional $O(\ln\frac{1}{\varepsilon})$ ancilla qubits [27]. Introducing classical feedforward process [26,38] can further reduce the number of ancilla to one without increasing circuit depth. Recent studies [5,6] utilize the structure of QSVT to reinterpret phase estimation methods, introducing potential trade-offs among precision, query complexity, and number of ancilla qubits. The quantum phase search (QPS) method proposed in this work is based on the intuitive idea of a binary search, which is fundamentally different from the previous QFT-based algorithms. Dong *et al.* [17] proposed a phase estimation algorithm, in a spirit similar to ours, for the case $U|\psi\rangle = e^{-i\lambda}|\psi\rangle$, where $\lambda \in [\eta, \pi - \eta]$ for some constant $\eta > 0$. However, this prior condition on the eigenvalue is usually not satisfied for an arbitrary unitary and initial state. Here we compare our phase search method to some previous phase estimation algorithms with respect to the query complexity and number of ancilla qubits with the same precision and success probability, which is shown in Table I. One can see that the quantum phase search method achieves the best query complexity while requiring the fewest ancilla qubits, turning out to be an efficient phase estimation algorithm.

IV. QUANTUM ENTROPY ESTIMATION

Quantum entropy is used to characterize the randomness and disorder of a quantum system, which has various theoretical and experimental applications of relevance. Estimating the entropy of a quantum system is an important problem in quantum information science. Classical methods of estimating the quantum entropies require the density matrix of a quantum state, which is costly, especially when the size of the system is large. Recent works proposed quantum algorithms that could efficiently estimate quantum entropies [11–13], showing potential quantum speedups over the classical methods. The motivating idea behind these quantum approaches is the purified quantum query model [7], which prepares a purification of a mixed state ρ. The purified query model can apply to cases where the states are generated by a quantum circuit and have applications in many tasks [11,40]. Formally, the purified quantum query oracle of a mixed state ρ is a unitary U_ρ acting

TABLE I. Comparison of QPE algorithms. In the query complexity, the $\tilde{O}$ notation omits log-log factors. The number of ancilla qubits is the total number of qubits used other for the system of U

Methods for QPE	Queries to controlled U	No. of ancilla qubits	Success probability	Precision
QFT based (from [26,27,39])	$O(\frac{1}{\delta}(1+\frac{1}{\varepsilon}))$	$\lceil \ln\frac{1}{\delta} + \ln(2+\frac{1}{\varepsilon}) \rceil$	$1-\varepsilon$	δ
Semiclassical QFT based (from [26,38])	$O(\frac{1}{\delta}(1+\frac{1}{\varepsilon}))$	1		
QSVT based (from [5,6])	$\tilde{O}(\frac{1}{\delta}\ln(\frac{1}{\varepsilon}))$	$\lceil \ln\frac{1}{\delta} \rceil + 3$ (or 3)		
QPP based (in Theorem 3)	$\tilde{O}(\frac{1}{\delta}\ln(\frac{1}{\varepsilon}))$	1		

as

$$U_\rho|0\rangle_A|0\rangle_B = |\Psi_\rho\rangle_{AB} = \sum_{j=0}^{2^n-1} \sqrt{p_j}|\psi_j\rangle_A|\phi_j\rangle_B \tag{13}$$

such that $\mathrm{tr}_B(|\Psi_\rho\rangle\langle\Psi_\rho|) = \rho$, where $\{|\psi_j\rangle\}$ and $\{|\phi_j\rangle\}$ are sets of orthonormal states on systems A and B, respectively. Using the qubitization method in [7], such an oracle model can be used to build a qubitized block encoding $\hat{U}_\rho$ of the target state ρ. We show the detailed construction of $\hat{U}_\rho$ in Appendix D 2.

Consequently, it is reasonable and compelling to investigate whether QPP, the structure designed for unitary phase processing, could contribute to the improvement of quantum algorithms for quantum entropy estimation. Note that quantum entropies of a quantum state ρ can be interpreted as the corresponding classical entropies of the eigenvalues of ρ. If one could find trigonometric polynomials $F(x)$ that approximate the classical entropic functions, then quantum entropies can be naturally estimated via phase evaluation of $\hat{U}_\rho$ in Theorem 2 by the spectral correspondence between ρ and $\hat{U}_\rho$. Specifically, the following theorem is the basic principle of the QPP-based quantum entropy estimation; its proof is deferred to Appendix F 1.

Theorem 4. Let $|\Psi_\rho\rangle_{AB}$ be a purification of an n-qubit state ρ and $\hat{U}_\sigma$ be a qubitized block encoding of an n-qubit state σ with m ancilla qubits. For any real-valued polynomial $f(x) = \sum_{k=0}^{L} c_j x^k$, with $|F(x)| \leqslant 1$ for all $x \in \mathbb{R}$, there exists a QPP circuit $V(\hat{U}_\sigma)$ of L layers such that

$$\langle Z^{(0)}\rangle_{|\Phi\rangle} = \mathrm{tr}[\rho f(\sigma)], \tag{14}$$

where $|\Phi\rangle = [V(\hat{U}_\sigma) \otimes I_B]|0^{\otimes(m+1)}\rangle|\Psi_\rho\rangle_{AB}$ and the polynomial on a quantum state is defined as $f(\sigma) = \sum_{k=0}^{L} c_j \sigma^k$.

Theorem 4 shows that one could measure the value of $\mathrm{tr}[\rho f(\sigma)]$ as the Z expectation value of the ancilla qubit. An estimate of the expectation value within an additive error δ can be obtained by measuring the ancilla qubit $O(1/\delta^2)$ times. Moreover, we could directly apply the iterative amplitude estimation [35] on QPP to achieve a quadratic speedup without using extra qubits. For clarity, we present the QPP circuit of quantum entropy estimation in Fig. 3. We also note that Theorem 4 can be applied to extract many information-theoretic properties of quantum states other than quantum entropies.

In this section, to demonstrate the power of QPP, we utilize the generic method in Theorem 4 to estimate the most fundamental entropic functionals for quantum systems, including the von Neumann entropy, the quantum relative entropy, and the family of quantum Rényi entropies.

A. von Neumann and quantum relative entropy estimation

The von Neumann entropy [41] is a generalization of the Shannon entropy from the classical information theory to quantum information theory. For an n-qubit quantum state ρ, the von Neumann entropy is defined as

$$S(\rho) = -\mathrm{tr}(\rho \ln \rho). \tag{15}$$

Letting $\{p_j\}_j$ be the eigenvalues of ρ, then the von Neumann entropy is the same as the Shannon entropy of the probability distribution $\{p_j\}_j$,

$$S(\rho) = -\sum_{j=0}^{2^n-1} p_j \ln p_j. \tag{16}$$

Recall from the qubitization technique that partial eigenphases of the qubitized block encoding $\hat{U}_\rho$ are given by $\pm\tau_j = \pm\arccos(p_j)$; then we have $p_j = \cos(\pm\tau_j) \in [0, 1]$. Here we assume that the nonzero eigenvalues are lower bounded by some $\gamma > 0$. Then, by Theorem 4, the main idea of using QPP to estimate the von Neumann entropy is to find a polynomial $f(x)$ that approximates the function $\ln(x)$ with some appropriate scale on the interval $[\gamma, 1]$, and $|f(x)| \leqslant 1$ for $x \in [-1, 1]$. In particular, the polynomial $f(x)$ could be obtained from the Taylor series of $\ln(x)$. The overall result is stated in the following theorem.

Theorem 5 (von Neumann entropy estimation). Given a purified quantum query oracle U_ρ of a state ρ whose nonzero eigenvalues are lower bounded by $\gamma > 0$, there exists an algorithm that estimates $S(\rho)$ up to precision ε with high probability by measuring a single qubit, querying U_ρ and $U_\rho^\dagger$, $O(\frac{1}{\gamma\varepsilon^2}\ln^2(\frac{1}{\gamma})\ln(\frac{\ln(1/\gamma)}{\varepsilon}))$ times. Moreover, using amplitude estimation improves the query complexity to $O(\frac{1}{\gamma\varepsilon}\ln(\frac{1}{\gamma})\ln(\frac{\ln(1/\gamma)}{\varepsilon}))$.

In particular, the dependence on γ can be translated to the rank (or dimension) of the density matrix, from which we have the following corollary.

Corollary 3. Given a purified quantum query oracle U_ρ of a state ρ whose rank is $\kappa \leqslant 2^n$, there exists an algorithm that estimates $S(\rho)$ up to precision ε with high probability by measuring a single qubit, querying U_ρ and $U_\rho^\dagger$, $O(\frac{\kappa}{\varepsilon^3}\ln^3(\frac{\kappa}{\varepsilon})\ln(\frac{1}{\varepsilon}))$ times. Moreover, using amplitude estimation improves the query complexity to $O(\frac{\kappa}{\varepsilon^2}\ln^2(\frac{\kappa}{\varepsilon})\ln(\frac{1}{\varepsilon}))$.

We defer the proofs to Appendix F 2. Note that the estimation of the quantum relative entropy between states ρ and σ, i.e.,

$$D(\rho \parallel \sigma) = -\mathrm{tr}(\rho \ln \sigma) - S(\rho), \tag{17}$$

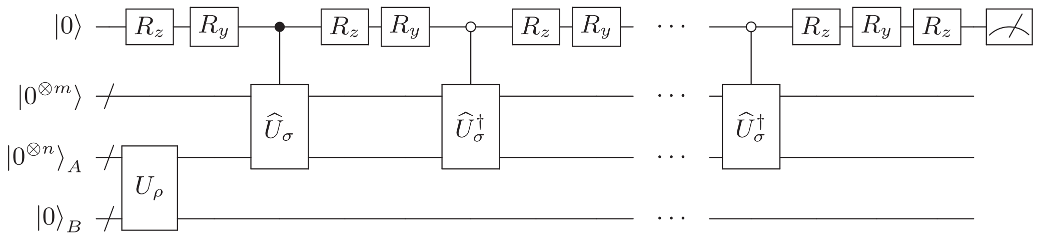


FIG. 3. The QPP circuit in Theorem 4 for quantum entropy estimation.

immediately follows from the above analysis. In particular, we only need to apply QPP on a qubitized block encoding of σ to estimate $\mathrm{tr}(\rho \ln \sigma)$ if the relative entropy is finite. The result of quantum relative entropy estimation is shown in Theorem 6.

Theorem 6 (quantum relative entropy estimation). Given purified quantum query oracles U_ρ and U_σ of states ρ and σ, respectively, such that their nonzero eigenvalues are lower bounded by $\gamma > 0$ and the kernel of σ has a trivial intersection with the support of ρ, there exists an algorithm that estimates $D(\rho \parallel \sigma)$ up to precision ε with high probability, querying U_ρ, U_σ, and their inverses $O(\frac{1}{\gamma\varepsilon^2}\ln^2(\frac{1}{\gamma})\ln(\frac{\ln(1/\gamma)}{\varepsilon}))$ times. Moreover, using amplitude estimation improves the query complexity to $O(\frac{1}{\gamma\varepsilon}\ln(\frac{1}{\gamma})\ln(\frac{\ln(1/\gamma)}{\varepsilon}))$.

B. Quantum Rényi entropy estimation

The quantum Rényi entropy [32] is a quantum version of the classical Rényi entropy that was first introduced in [42]. For $\alpha \in (0,1) \cup (1,\infty)$, the quantum α-Rényi entropy of an n-qubit quantum state ρ is defined as follows:

$$S_\alpha(\rho) = \frac{1}{1-\alpha}\ln \mathrm{tr}(\rho^\alpha). \tag{18}$$

Leting $\{p_j\}_j$ be the eigenvalues of ρ, then the quantum α-Rényi entropy reduces to the α-Rényi entropy of the probability distribution $\{p_j\}_j$,

$$S_\alpha(\rho) = \frac{1}{1-\alpha}\ln\left(\sum_{j=0}^{2^n-1} p_j^\alpha\right). \tag{19}$$

Similarly, we assume all nonzero eigenvalues are greater than some $\gamma > 0$. The method of Rényi entropy estimation, based on Theorem 4, is in the same spirit as estimating the von Neumann entropy; the only difference is that we now aim to find a polynomial $f(x)$ that approximates the function $x^{\alpha-1}$ for any $\alpha > 0$ and $\alpha \neq 1$ on the interval $[\gamma, 1]$. The exponent is $\alpha - 1$ because we can write $\mathrm{tr}(\rho^\alpha)$ as $\mathrm{tr}(\rho\rho^{\alpha-1})$, and the isolated ρ comes from the input state of QPP.

When $\alpha > 0$ is a noninteger, the polynomial could be given by separately considering the integer part and decimal part of $\alpha - 1$. Thus we only need to find a polynomial that approximates $x^{\alpha-1}$ for $\alpha \in (0,1)$ on the interval $x \in [\gamma, 1]$, which could be obtained from the Taylor series of $x^{\alpha-1}$. We present the results in the following theorem and further discussion in Appendix F 3.

Theorem 7 (quantum Rényi entropy estimation for real α). Given a purified quantum query oracle U_ρ of a state ρ whose nonzero eigenvalues are lower bounded by $\gamma > 0$, there exists an algorithm that estimates $S_\alpha(\rho)$ up to precision ε with high probability by measuring a single qubit, querying U_ρ and $U_\rho^\dagger$ a total of times of $\tilde{O}(\frac{1}{\gamma^{3-2\alpha}\varepsilon^2}\eta^2)$ if $\alpha \in (0,1)$ and $\tilde{O}(\frac{\alpha\gamma+1}{\gamma\varepsilon^2}\eta^2)$ if $\alpha \in (1,\infty)$, where $\eta = \frac{\mathrm{tr}(\rho^\alpha)^{-1}}{|1-\alpha|}$. Moreover, using quantum amplitude estimation improves the query complexity to $\tilde{O}(\frac{1}{\gamma^{2-\alpha}\varepsilon}\eta)$ if $\alpha \in (0,1)$ and $\tilde{O}(\frac{\alpha\gamma+1}{\gamma\varepsilon}\eta)$ if $\alpha \in (1,\infty)$. Here the $\tilde{O}$ notation omits logarithmic factors.

Similarly, we provide a method to estimate $S_\alpha(\rho)$ without information of γ in Appendix F 3. When α is an integer, the function $x^{\alpha-1}$ naturally becomes a normalized polynomial so that approximation error does not exist by Theorem 4. In this case, the dependence on the threshold γ can be further improved.

Theorem 8 (quantum Rényi entropy estimation for integer α). Suppose $\alpha > 1$ is a positive integer. Then there exists an algorithm that estimates $S_\alpha(\rho)$ up to precision ε with high probability by measuring a single qubit, querying U_ρ and $U_\rho^\dagger$, $O(\frac{\alpha\,\mathrm{tr}(\rho^\alpha)^{-2}}{\varepsilon^2})$ times. Moreover, using amplitude estimation improves the query complexity to $O(\frac{\alpha\,\mathrm{tr}(\rho^\alpha)^{-1}}{\varepsilon})$.

Note that this method of computing $S_\alpha(\rho)$ for $\alpha \in \mathbb{N}_+$ naturally establishes an efficient algorithm for entanglement spectroscopy, a task of obtaining the entanglement of a quantum state. Consider a bipartite pure state $|\Psi_\rho\rangle_{AB}$ in Eq. (13). The entanglement between systems A and B can be characterized by the eigenvalues of the reduced density operator $\rho = \mathrm{tr}_B(|\Psi_\rho\rangle\langle\Psi_\rho|)$. Specifically, one needs to compute the $\mathrm{tr}(\rho^k)$ for $k = 1, \ldots, k_{\max}$ to estimate $k_{\max}$ largest eigenvalues of ρ by the Newton-Girard method [43–45].

C. Comparison to related works

As introduced above, we utilize the structure of QPP to estimate quantum entropies based on the purified quantum query model. Here we briefly mention some closely related works on quantum entropy estimation under a similar setting and provide the comparison of algorithms in Table II. For von Neumann entropy $S(\rho)$, Gilyén and Li [11] proposed an efficient quantum algorithm based on QSVT and amplitude estimation that achieves a near-linear query complexity and an additive error ε. Work by Gur *et al.* [13] utilized the quantum singular value estimation [46] and amplitude estimation to implement an algorithm with a sublinear query complexity up to a multiplicative error bound. By contrast, our algorithm could estimate the result by measuring the first ancilla qubit, which has a slightly worse query complexity in the worst case but a smaller circuit size. Nevertheless, by Corollary 3, our complexity can depend on the rank of the density matrix, which will be further improved for low-rank cases. Note that one

TABLE II. Comparison of algorithms for estimating von Neumann entropy within additive error. Here the $\tilde{O}$ notation omits logarithmic factors, $\gamma > 0$ is the lower bound of eigenvalues, $\kappa > 0$ is the rank of the state $\rho \in \mathbb{C}^{d\times d}$, and ε is the additive error of estimating $S(\rho)$. Here and in the following tables QAE is short for quantum amplitude estimation.

Methods for $S(\rho)$ estimation	Total queries to U_ρ and $U_\rho^\dagger$	Queries per use of circuit
QSVT based with QAE (from [11])	$\tilde{O}(\frac{d}{\varepsilon^{1.5}})$	$\tilde{O}(\frac{d}{\varepsilon^{1.5}})$
QPP based (assumes rank, in Corollary 3)	$\tilde{O}(\frac{\kappa}{\varepsilon^3})$	$\tilde{O}(\frac{\kappa}{\varepsilon})$
QPP based with QAE (assumes rank, in Corollary 3)	$\tilde{O}(\frac{\kappa}{\varepsilon^2})$	$\tilde{O}(\frac{\kappa}{\varepsilon^2})$
QPP based (in Theorem 5)	$\tilde{O}(\frac{1}{\gamma\varepsilon^2})$	$\tilde{O}(\frac{1}{\gamma})$
QPP based with QAE (in Theorem 5)	$\tilde{O}(\frac{1}{\gamma\varepsilon})$	$\tilde{O}(\frac{1}{\gamma\varepsilon})$

could apply amplitude estimation on QPP without using extra qubits, which might be required for previous QSVT-based algorithms. As a result, the QPP-based algorithms allow us to flexibly consider the trade-off between the query complexity and the circuit depth in practical applications.

With regard to the family of quantum α-Rényi entropies $S_\alpha(\rho)$, when α is an integer, QPP establishes an efficient algorithm for entanglement spectroscopy. Compared to previous algorithms for entanglement spectroscopy [43,44], the QPP-based algorithm significantly reduces the circuit width from $\Theta(n\alpha)$ to $4n + 1$ without using qubit resets as in [45]. For a more general case that α is not an integer, Subramanian and Hsieh [12] introduced a quantum algorithm that combines the QSVT technique and the deterministic quantum computation with one clean qubit (DQC1) method. Their algorithm estimates $S_\alpha(\rho)$ with an additive error by measuring a single qubit, using an expected total of $\tilde{O}(d^2/\gamma\varepsilon^2)$ queries to the purified quantum oracle, where $d = 2^n$ is the dimension of ρ. Our QPP-based approaches improve the results in [12] in terms of the dependence on the dimension. For instance, for $\alpha > 1$, our algorithms based on single-qubit measurement require a query complexity of $\tilde{O}(\frac{\alpha}{\gamma\varepsilon^2})$. The main reason there is a speedup factor of d^2 is that the DQC1 method requires a maximally mixed state $\frac{1}{d}I$ as the input state, whereas the QPP-based method uses ρ as the input state. Moreover, as we mentioned before, QSVT is not natively compatible with indirect measurement; one needs to utilize an extra ancilla qubit to control the QSVT circuit in order to implement indirect measurement, as shown in [12]. Similar to the earlier description, we could leverage quantum amplitude estimation to achieve a better query complexity. More detailed comparisons are shown in Table III.

The polynomial transformation implemented by QSVT lies in the amplitude of the outcome state, which could not be obtained by indirect measurements of the ancilla qubit in QSVT. Thus most QSVT-based entropy estimation algorithms estimate the value by either applying amplitude estimation or combining with the DQC1 model, and both of these methods increase the circuit size. Another approach is using a polynomial to estimate the square root of the function $\sqrt{f(x)}$, as shown by Wang *et al.* [47]. This approach makes QSVT compatible with indirect measurements, since the approximated function now can be represented by the probability of measuring, like in QPP. However, the problem is that sometimes $\sqrt{f(x)}$ could be more difficult to approximate than $f(x)$. For example, $f(x) = x$ is just a simple one-term polynomial, whereas $\sqrt{f(x)} = \sqrt{x}$ takes many more terms to precisely approximate. Thus such a method presented in [47] may lead to even worse complexity than previous ones in [11–13]. We defer further discussion and comparison to Appendix F 4.

V. HAMILTONIAN SIMULATION

This section aims to investigate the application of our phase processing circuits in simulating the dynamics of a quantum system. Unsurprisingly, our algorithms can reproduce the optimal results introduced in [19]. Specifically, for the Hamiltonian H, given an evolution time t and a simulation error ε, the goal is to simulate the time evolution e^{-iHt} with error ε, i.e., produce a unitary U such that $\|U - e^{-iHt}\| \leqslant \varepsilon$. Usually, this is accomplished by designing a quantum circuit to simulate the operator e^{-iHt} with high precision. In the setup, we assume a block encoding of the Hamiltonian, which is a commonly used input model appearing in [4,7,48].

TABLE III. Comparison of algorithms for estimating quantum α-Rényi entropies within additive error for different α. Here the $\tilde{O}$ notation omits logarithmic factors, $\gamma > 0$ is the lower bound of eigenvalues of a mixed state $\rho \in \mathbb{C}^{d\times d}$, $\eta := \frac{\mathrm{tr}(\rho^\alpha)^{-1}}{|1-\alpha|}$ is the quantity depending on α and ρ, and ε is the additive error of estimating $S_\alpha(\rho)$.

	Total queries to U_ρ and $U_\rho^\dagger$		
Methods for $S_\alpha(\rho)$ estimation	$\alpha \in (0, 1)$	$\alpha \in (1, \infty)$	$\alpha \in \mathbb{N}_+$
QSVT based with DQC1 (from [12])	$\tilde{O}(\frac{d^2}{\gamma\varepsilon^2}\eta^2)$	$\tilde{O}(\frac{d^2}{\gamma\varepsilon^2}\eta^2)$	$O(\frac{d^2}{\varepsilon^2}\alpha^3\eta^2)$
QPP based (in Theorems 7 and 8)	$\tilde{O}(\frac{1}{\gamma^{3-2\alpha}\varepsilon^2}\eta^2)$	$\tilde{O}(\frac{\alpha\gamma+1}{\gamma\varepsilon^2}\eta^2)$	$O(\frac{1}{\varepsilon^2}\alpha\eta^2)$
QPP based with QAE (in Theorems 7 and 8)	$\tilde{O}(\frac{1}{\gamma^{2-\alpha}\varepsilon}\eta)$	$\tilde{O}(\frac{\alpha\gamma+1}{\gamma\varepsilon}\eta)$	$O(\frac{1}{\varepsilon}\alpha\eta)$

Before presenting the results, we recall the definition of block encoding.

A. Block encoding

A block encoding of a matrix $A \in \mathbb{C}^{2^n \times 2^n}$ with spectral norm $\|A\| \leqslant 1$ is a unitary matrix U_A such that the upper left block of the matrix is A, i.e.,

$$U_A = \begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}. \tag{20}$$

Then we can write $A = (\langle 0^{\otimes m}| \otimes I^{\otimes n})U_A(|0^{\otimes m}\rangle \otimes I^{\otimes n})$, where m denotes the number of ancilla qubits in the block encoding. In other words, for any state $|\psi\rangle \in \mathbb{C}^N$ we have $(\langle 0^{\otimes m}| \otimes I^{\otimes n})U_A|0^{\otimes m}, \psi\rangle = A|\psi\rangle$. In particular, letting $|\psi_\lambda\rangle$ be an eigenvector of A with an eigenvalue λ, then we will have a state

$$U_A|0^{\otimes m}, \psi_\lambda\rangle = \lambda|0^{\otimes m}, \psi_\lambda\rangle + \sqrt{1-\lambda^2}|\bot_\lambda\rangle, \tag{21}$$

where $|\bot_\lambda\rangle$ denotes a state orthogonal to $|0^{\otimes m}, \psi_\lambda\rangle$. In this equation, we absorb the relative phase into states and ignore the global phase. To associate the spectrum of A and its block encoding U, it has to be ensured that the subspace $\text{span}\{|0^{\otimes m}, \psi_\lambda\rangle, U|0^{\otimes m}, \psi_\lambda\rangle\}$ is invariant under U_A. However, this is generally not true for an arbitrary block encoding. To resolve this issue, Low and Chuang [7] proposed a so-called qubitization technique that uses controlled U_A and controlled $U_A^\dagger$ once to implement a qubitized block encoding $\hat{U}_A$ preserving the subspace $\text{span}\{|0^{\otimes(m+1)}, \psi_\lambda\rangle, \hat{U}_A|0^{\otimes(m+1)}, \psi_\lambda\rangle\}$. Write

$$\hat{U}_A|0^{\otimes(m+1)}, \psi_\lambda\rangle = \lambda|0^{\otimes(m+1)}, \psi_\lambda\rangle + \sqrt{1-\lambda^2}|\hat{\bot}_\lambda\rangle, \tag{22}$$

where $|\hat{\bot}_\lambda\rangle$ denotes a state orthogonal to $|0^{\otimes(m+1)}, \psi_\lambda\rangle$. Then

$$|\chi_\lambda^\pm\rangle = \frac{1}{\sqrt{2}}(|0^{\otimes(m+1)}, \psi_\lambda\rangle \pm i|\hat{\bot}_\lambda\rangle) \tag{23}$$

are eigenstates of $\hat{U}_A$ with eigenphases $\pm\tau_\lambda = \pm \arccos \lambda$. Now the spectrum of A and that of $\hat{U}_A$ are associated, which allows us to process and extract the spectrum of an arbitrary matrix A by applying QPP on $\hat{U}_A$. More details of qubitization are discussed in Appendix D 1. In the next section, we will show how to use QPP to solve the Hamiltonian simulation problem.

B. Hamiltonian simulation

Suppose we have a block encoding $U_H = \left[\begin{smallmatrix} H/\Lambda & \\ \cdot & \end{smallmatrix}\right]$. With out loss of generality, we may assume $\|H\| \leqslant 1$ and $\Lambda = 1$, since for $\|H\| > 1$ the problem can be considered as simulating evolution under the rescaled Hamiltonian H/Λ for time Λt. Recall that the qubitization establishes the relation between the eigenvalues of the Hamiltonian H and eigenphases of its qubitized block encoding $\hat{U}_H$, i.e., $\tau_\lambda = \arccos(\lambda)$. Since the time-evolution operator e^{-iHt} can be decomposed as $e^{-i\lambda t}$, the main idea is to transform eigenphases of $\hat{U}_H$ by Theorem 1 as $\tau_\lambda \mapsto e^{-i\lambda t}$. We select a trigonometric polynomial $F(x)$ that approximates the function $f(x) = e^{-i\cos(x)t}$ with desired precision. Then applying the trigonometric polynomial $F(x)$ on each eigenphase τ_λ approximates

$$f(\tau_\lambda) = e^{-i\cos(\tau_\lambda)t} = e^{-i\lambda t}, \tag{24}$$

which provides a precise approximation of the time-evolution operator e^{-iHt}. The query complexity of Hamiltonian simulation is analyzed in the following theorem, and the detailed proof is deferred to Appendix E.

Theorem 9. Given a block encoding U_H of H/Λ for some $\Lambda \geqslant \|H\|$, there exists an algorithm that simulates evolution under the Hamiltonian H for time $t \in \mathbb{R}$ within precision $\delta > 0$, using two ancilla qubits and querying controlled U_H and controlled $U_H^\dagger$ for a total number of times in

$$\Theta\left(\Lambda|t| + \frac{\ln(2/\delta^2)}{\ln\left(e + \frac{\ln(2/\delta^2)}{\Lambda|t|}\right)}\right).$$

We remark that one could measure the ancilla qubit after applying our circuit, and the residual state is highly approximate to the desired state. In this case, we can relax the function approximation error. The final state approximates the target state with an error of δ, succeeding with a probability at least $1 - 2\delta$. As a result, the circuit depth can be reduced.

From the result, we conclude that QPP solves Hamiltonian simulation problems with the access to block encoding, and the query complexity matches the optimal results as in [7]. We notice that QPP could also solve other Hamiltonian problems such as spectrum estimation, ground-state energy estimation, and ground-state preparation. Details are in Appendix E.

VI. CONCLUSION

In this paper, we introduced a quantum phase processing algorithmic toolbox based on applying trigonometric transformations to eigenphases of a unitary operator. The toolbox allows us to implement a desired evolution to the input state and, more interestingly, to extract the eigeninformation of quantum systems by simply measuring the ancilla qubit. We demonstrated the capability of this toolbox by developing QPP-based algorithms for solving a variety of problems. Owing to the capability of QPP to directly process eigenphases, we designed an efficient and intuitive phase estimation algorithm purely based on the idea of binary search, which uses only one ancilla qubit and matches the best prior performance. Utilizing block encoding and qubitization, we showed that QPP could reproduce and even improve previous quantum algorithms based on the framework of QSVT, such as Hamiltonian simulation and quantum entropy estimation.

Quantum phase processing generalizes the trigonometric QSP by extending the R_z rotation instead of the reflection in the Chebyshev-based QSP as QSVT did, providing an alternative interpretation for applying QSP to multiple qubits. Due to the common underlying logic between these two QSP conventions, it is unavoidable to find similarities between QPP and QSVT. However, due to the distinctions between trigonometric and Chebyshev-based QSP, our results show that QPP is a powerful toolbox for developing quantum algorithms related to eigenphase transformation and processing, which essentially complements the existing QSVT framework. On one hand, QPP implements arbitrary complex trigonometric polynomial, which overcomes the parity constraints in QSVT and thus exempts one from using a linear combination of unitaries in certain cases. On the other hand, QPP is natively compatible with indirect measurements, which could extract eigeninformation of the main system via measuring the single ancilla qubit. Notably, QPP could work without amplitude estimation, which requires shorter circuits and less coherence time than QSVT and hence might be more friendly to near-term quantum hardware. Quantum phase processing natively inherits the trick of phase kickback and thus it is particularly suitable for developing phase-related quantum algorithms. Other than applications mentioned in this paper, QPP can be potentially applied to other problems, including but not limited to Rényi divergence estimation, unitary trace estimation, and quantum Monte Carlo method. Moreover, considering the connections between quantum signal processing and single-qubit quantum neural networks [25,49], our results may provide insight and lead to applications of QSVT and QPP in the area of quantum machine learning [50–59]. Overall, we believe that the QPP algorithmic toolbox would deepen our understanding of quantum algorithm design and be useful in the search for quantum applications in quantum physics, quantum chemistry, machine learning, and beyond.

ACKNOWLEDGMENTS

Part of this work was done when Y.W., L.Z., Z.Y., and X.W. were at Baidu Research. X.W. acknowledges support from the Start-up Fund from the Hong Kong University of Science and Technology (Guangzhou). Y.W. acknowledges support from the Baidu-UTS AI Meets Quantum project, the National Natural Science Foundation of China (Grant No. 62071240), and the Innovation Program for Quantum Science and Technology (Grant No. 2021ZD0302900).

APPENDIX A: BRIEF SUMMARY OF COMMON QSP CONVENTIONS

As shown in Table IV, the trigonometric QSP can use a single-qubit quantum system to make an arbitrary transformation of a polynomial in $\mathbb{C}[e^{ix/2}, e^{-ix/2}]$ with parity. However, since all normalized and square-integrable functions can be approximated by the polynomials in $\mathbb{C}[e^{ix/2}, e^{-ix/2}]$ with parity 0, i.e., the polynomials in $\mathbb{C}[e^{ix}, e^{-ix}]$, the trigonometric QSP can be claimed to be parity-free.

APPENDIX B: THEOREMS OF QUANTUM PHASE PROCESSING

1. Existence of Laurent complementary and angle finding

Other than the characterization of trigonometric QSP in Lemma 1, we also need to specify the achievable trigonometric polynomials $P(x)$ for which there exists a corresponding $Q(x)$ satisfying the three conditions in Lemma 1. First we prove the following lemma, using similar ideas of root finding from previous works [23,24,33].

Lemma 4. Suppose $A(x)$ is a non-negative real-valued trigonometric polynomial. Then there exists a Laurent polynomial $Q \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ such that $QQ^* = A$.

Proof. Define $A(x) = \sum_{j=-2L}^{2L} a_j e^{ijx/2}$. We can decompose A by the set of $4L$ roots $\{\xi_k\}_{k=1}^{4L}$ so that

$$A(x) = a_{2L} e^{-iLx} \prod_{k=1}^{4L} (e^{ix/2} - \xi_k), \tag{B1}$$

where $\{\xi_k\}_{k=1}^{4L}$ can be efficiently found by computing all roots of a regular complex polynomial $g(\xi) := \sum_{j=0}^{4L} a_{j-2L}\xi^j$. In particular, since A is real and non-negative, these roots appear in inverse conjugate pairs, i.e., $\{\xi_k\}_{k=1}^{4L} = \{\xi_k, \frac{1}{\xi_k^*}\}_{k=1}^{2L}$. Then A can be further simplified to

$$A(x) = a_{2L} e^{-iLx} \left(\prod_{k=1}^{2L} (e^{ix/2} - \xi_k) \right) \left[\prod_{k=1}^{2L} \left(e^{ix/2} - \frac{1}{\xi_k^*} \right) \right]. \tag{B2}$$

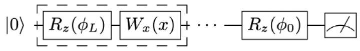
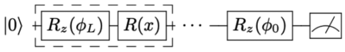
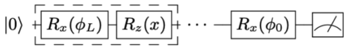
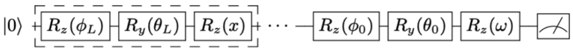
From the fact $e^{ix/2} - \xi_k = -e^{ix/2}\xi_k(e^{-ix/2} - \frac{1}{\xi_k})$, we have

$$A(x) = a_{2L} \left(\prod_{k=1}^{2L} \xi_k \right) \left[\prod_{k=1}^{2L} \left(e^{-ix/2} - \frac{1}{\xi_k} \right) \right] \times \left[\prod_{k=1}^{2L} \left(e^{ix/2} - \frac{1}{\xi_k^*} \right) \right]. \tag{B3}$$

Let $q := a_{2L} \prod_{k=1}^{2L} \xi_k$. Note that q is real since

$$A(0) = a_{2L} g(0) = a_{2L} \prod_{k=1}^{2L} \frac{\xi_k}{\xi_k^*} = \frac{a_{2L} \left(\prod_{k=1}^{2L} \xi_k \right)^2}{\prod_{k=1}^{2L} |\xi_k|^2} \tag{B4}$$

TABLE IV. Summary of the quantum circuits and the corresponding types of realizable polynomials retrieved by various conventions of QSP, most of them originally done by Martyn *et al.* [5] in their Appendix A. Note that $W_x(x) := \begin{bmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{bmatrix}$, $R(x) := \begin{bmatrix} x & \sqrt{1-x^2} \\ \sqrt{1-x^2} & -x \end{bmatrix}$, and all polynomials are assumed to be normalized.

Conventions of QSP	QSP circuits	Type of polynomials
W_x (from [18])		$\mathbb{C}[x]$ with parity L mod2 and other constraints
Reflection (from [4])		$\mathbb{C}[x]$ with parity L mod2 and other constraints
W_z (from [23,24]) or trigonometric		$\mathbb{R}[e^{ix/2}, e^{-ix/2}]$ with parity L mod2
Improved trigonometric (from [25])	 $\|0\rangle$ $R_z(\phi_L)$ $R_y(\theta_L)$ $R_z(x)$ $R_z(\phi_0)$ $R_y(\theta_0)$ $R_z(\omega)$	$\mathbb{C}[e^{ix/2}, e^{-ix/2}]$ with parity L mod2

is real. Define $Q(x) := \sqrt{q}e^{-iLx/2}\prod_{k=1}^{2L}(e^{ix/2} - \frac{1}{\xi_k^*})$ and the result follows. ■

Using Lemma 4, we show that there exists a Q such that $QQ^* = 1 - PP^*$ for any trigonometric series P satisfying $|P|^2 \leqslant 1$.

Lemma 5 (existence of Laurent complementary). Let $P \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ be a Laurent polynomial with degree no larger than L and parity $L \bmod 2$. Then $|P(x)| \leqslant 1$ for all $x \in \mathbb{R}$ if and only if there exists a Laurent polynomial $Q \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ satisfying the following conditions: (a) $\deg(Q) \leqslant L$, (b) Q has parity $L \bmod 2$, and (c) $|P(x)|^2 + |Q(x)|^2 = 1\,\forall\, x \in \mathbb{R}$.

Proof. ($\Longleftarrow$) The statement trivially holds from condition (c) $|P(x)|^2 + |Q(x)|^2 = 1$.

($\Longrightarrow$) Suppose $P \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ is a Laurent polynomial satisfying the above requirements. Note that if $|P(x)| = 1$ for all $x \in \mathbb{R}$, then the result follows by setting $Q = 0$. Suppose $|P(x)| < 1$ for some $x \in \mathbb{R}$. Then $A = 1 - PP^*$ is a real and positive Laurent polynomial. By Lemma 4 there exists a Laurent polynomial $Q \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ such that $QQ^* = 1 - PP^*$, i.e., $|P(x)|^2 + |Q(x)|^2 = 1$ for all $x \in \mathbb{R}$. Conditions (a) and (b) follow from the fact that $\deg(P) \leqslant L$ and P has parity $L \bmod 2$. ■

Combining Lemmas 5 and 1, now we can compute the corresponding rotation angles α, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$ of $W_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}$ for a given trigonometric polynomial $P(x)$, as shown in Algorithm 3.

Algorithm 3. Angle finding.

```
Require: A Laurent polynomial $P \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ such that
    $|P(x)| \leqslant 1$ for all $x \in \mathbb{R}$.
Ensure: Rotation parameters $\omega$, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$ such that
    $\langle 0|W_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)|0\rangle = P(x)$ for all $x \in [-\pi, \pi]$.
Compute the set of roots $\{\xi_k\}_{k=1}^{4L}$ and leading coefficient $a_{2L}$
  from real-valued and non-negative trigonometric polynomial
  $A(x) = 1 - P(x)P^*(x)$, where $L$ is the degree of $P$. Sort the
  set by ascending modulus and determine
    $Q \leftarrow \sqrt{a_{2L}\prod_{k=1}^{2L}\xi_k}e^{-iLx/2}\prod_{k=1}^{2L}(e^{ix/2} - \frac{1}{\xi_k^*})$.
while $\deg(P) > 0$ do
    $k \leftarrow \deg(P)$, $p_k \leftarrow P[e^{ikx/2}]$, $p_{-k} \leftarrow P[e^{-ikx/2}]$,
  $q_k \leftarrow Q[e^{ikx/2}]$, and $q_{-k} \leftarrow Q[e^{-ikx/2}]$.
  Determine $\theta_k, \phi_k$ such that $\cos\frac{\theta_k}{2}e^{i\phi_k/2}p_k + \sin\frac{\theta_k}{2}e^{-i\phi_k/2}q_k = 0$
  and $-\sin\frac{\theta_k}{2}e^{i\phi_k/2}p_{-k} + \cos\frac{\theta_k}{2}e^{-i\phi_k/2}q_{-k} = 0$.
  Update polynomials (simultaneously) such that
    $P \leftarrow e^{i\phi_k/2}\cos\frac{\theta_k}{2}e^{ix/2}P + e^{-i\phi_k/2}\sin\frac{\theta_k}{2}e^{ix/2}Q$ and
    $Q \leftarrow e^{-i\phi_k/2}\cos\frac{\theta_k}{2}e^{-ix/2}Q - e^{i\phi_k/2}\sin\frac{\theta_k}{2}e^{-ix/2}P$.
end while
Determine $\omega$, $\theta_0$, and $\phi_0$ such that
  $R_z(\omega)R_y(\theta_0)R_z(\phi_0) = \begin{bmatrix} P & -Q \\ Q^* & P^* \end{bmatrix}$, where $\boldsymbol{\theta} \leftarrow (\theta_0, \dots, \theta_L)$
  and $\boldsymbol{\phi} \leftarrow (\phi_0, \dots, \phi_L)$.
Return $\omega$, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$.
```

2. Proofs of Corollaries 1 and 2

The following is the proof of Corollary 1.

Proof. Note that F is a Laurent polynomial with degree no larger than $2L$. By Lemma 5 there exists a Laurent polynomial $G \in \mathbb{C}[e^{ix/2}, e^{-ix/2}]$ such that $\deg(G) \leqslant 2L$, G has parity 0, and $|F(x)|^2 + |G(x)|^2 = 1$ for all $x \in \mathbb{R}$. By Lemma 1 there exists $\omega \in \mathbb{R}$, $\boldsymbol{\theta} \in \mathbb{R}^{2L+1}$, and $\boldsymbol{\phi} \in \mathbb{R}^{2L+1}$ such that

$$W_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}^{2L} = \begin{bmatrix} F & -G \\ G^* & F^* \end{bmatrix}. \tag{B5}$$

The result directly follows. ■

The following is the proof of Corollary 2.

Proof. Note that $[1 \pm F(x)]/2$ are non-negative real-valued trigonometric polynomials. Then by Lemma 4 there exist two Laurent polynomials P and Q such that for all $x \in \mathbb{R}$,

$$P(x)P^*(x) = \frac{1 + F(x)}{2}, \quad Q(x)Q^*(x) = \frac{1 - F(x)}{2}. \tag{B6}$$

Observe that P and Q satisfy the three conditions in Lemma 1; thus there exist $\omega \in \mathbb{R}$, $\boldsymbol{\theta} \in \mathbb{R}^{L+1}$, and $\boldsymbol{\phi} \in \mathbb{R}^{L+1}$ such that

$$\begin{aligned}\langle 0|W_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}^{L}(x)^\dagger Z W_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}^{L}(x)|0\rangle \\ = P(x)P^*(x) - Q(x)Q^*(x) = F(x)\end{aligned} \tag{B7}$$

for all $x \in \mathbb{R}$. ■

3. Proof of Lemma 2

The following is the proof of Lemma 2.

Proof. Observe that the decomposition of unitaries is

$$\begin{bmatrix} U^\dagger & 0 \\ 0 & I \end{bmatrix} = \sum_{j=0}^{2^n-1} e^{-i\tau_j}|0\rangle\langle 0| \otimes |\chi_j\rangle\langle\chi_j| + \sum_{j=0}^{2^n-1}|1\rangle\langle 1| \otimes |\chi_j\rangle\langle\chi_j| \tag{B8}$$

$$= \sum_{j=0}^{2^n-1} e^{-i\tau_j/2}\begin{bmatrix} e^{-i\tau_j/2} & 0 \\ 0 & e^{i\tau_j/2} \end{bmatrix} \otimes |\chi_j\rangle\langle\chi_j| = \sum_{j=0}^{2^n-1} e^{-i\tau_j/2}R_z(\tau_j) \otimes |\chi_j\rangle\langle\chi_j| \tag{B9}$$

$$= \bigoplus_{j=0}^{2^n-1} e^{-i\tau_j/2}R_z(\tau_j)_{\mathbb{B}_j}, \tag{B10}$$

$$\begin{bmatrix} I & 0 \\ 0 & U \end{bmatrix} = \sum_{j=0}^{2^n-1}|0\rangle\langle 0| \otimes |\chi_j\rangle\langle\chi_j| + \sum_{j=0}^{2^n-1} e^{i\tau_j}|1\rangle\langle 1| \otimes |\chi_j\rangle\langle\chi_j| \tag{B11}$$

$$= \sum_{j=0}^{2^n-1} e^{i\tau_j/2}\begin{bmatrix} e^{-i\tau_j/2} & 0 \\ 0 & e^{i\tau_j/2}\end{bmatrix} \otimes |\chi_j\rangle\langle\chi_j| = \sum_{j=0}^{2^n-1} e^{i\tau_j/2} R_z(\tau_j) \otimes |\chi_j\rangle\langle\chi_j| \tag{B12}$$

$$= \bigoplus_{j=0}^{2^n-1} e^{i\tau_j/2} R_z(\tau_j)_{\mathbb{B}_j}. \tag{B13}$$

In a similar manner, we can decompose R_y and R_z gates applied on the first qubit such that for any $\zeta \in \mathbb{R}$,

$$R_y^{(0)}(\zeta) \otimes I = R_y^{(0)}(\zeta) \otimes \sum_{j=0}^{2^n-1} |\chi_j\rangle\langle\chi_j| = R_y^{(0)}(\zeta) \otimes \bigoplus_{j=0}^{2^n-1} 1_{\{|\chi_j\rangle\}} = \bigoplus_{j=0}^{2^n-1} R_y(\zeta)_{\mathbb{B}_j},$$

$$R_z^{(0)}(\zeta) \otimes I = \cdots = \bigoplus_{j=0}^{2^n-1} R_z(\zeta)_{\mathbb{B}_j}. \tag{B14}$$

For convenience, the $\mathbb{B}_j$ are omitted in the rest of the proof. From the above equations, for all even $L \in \mathbb{N}$, $\omega \in \mathbb{R}$, and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{L+1}$,

$$V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U) = \bigoplus_{j=0}^{2^n-1} R_z(\omega)R_y(\theta_0)R_z(\phi_0)\prod_{l=1}^{L/2} e^{-i\tau_j/2}R_z(\tau_j)R_y(\theta_{2l-1})R_z(\phi_{2l-1})e^{i\tau_j/2}R_z(\tau_j)R_y(\theta_{2l})R_z(\phi_{2l}) \tag{B15}$$

$$= \bigoplus_{j=0}^{2^n-1} R_y(\theta_0)R_z(\phi_0)\prod_{l=1}^{L} R_z(\tau_j)R_y(\theta_l)R_z(\phi_l) = \bigoplus_{j=0}^{2^n-1} W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(\tau_j). \tag{B16}$$

A similar statement holds for odd $L \in \mathbb{N}$. ■

4. Proofs of Theorems 1 and 2

The following is the proof of Theorem 1.

Proof. By Corollary 1 there exists $\omega \in \mathbb{R}$, $\boldsymbol{\theta} \in \mathbb{R}^{2L+1}$, and $\boldsymbol{\phi} \in \mathbb{R}^{2L+1}$ such that $\langle 0|W^{2L}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)|0\rangle = F(x)$. Then for such ω, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$,

$$(\langle 0| \otimes I^{\otimes n})V^{2L}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)(|0\rangle \otimes I^{\otimes n}) = \sum_{j=0}^{2^n-1}(\langle 0| \otimes I^{\otimes n})V^{2L}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)(|0\rangle \otimes |\chi_j\rangle\langle\chi_j|) \tag{B17}$$

$$= \sum_{j=0}^{2^n-1}[(\langle 0| \otimes I^{\otimes n})V^{2L}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)|0, \chi_j\rangle]\langle\chi_j| \tag{B18}$$

$$= \sum_{j=0}^{2^n-1}\langle 0|_{\mathbb{B}_j} W^{2L}_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)_{\mathbb{B}_j}(\tau_j)|0\rangle_{\mathbb{B}_j}\langle\chi_j| \quad \text{(by Lemma 2)} \tag{B19}$$

$$= \sum_{j=0}^{2^n-1} F(\tau_j)|\chi_j\rangle\langle\chi_j| \tag{B20}$$

$$= F(U), \tag{B21}$$

as required. ■

The following is the proof of Theorem 2.

Proof. We begin the proof by decomposing the observable $Z^{(0)} \otimes I$. Note that

$$Z^{(0)} \otimes I = Z^{(0)} \otimes \sum_{j=0}^{2^n-1} |\chi_j\rangle\langle\chi_j| = \bigoplus_{j=0}^{2^n-1} Z_{\mathbb{B}_j}, \tag{B22}$$

where $\mathbb{B}_j = \{|0, \chi_j\rangle, |1, \chi_j\rangle\}$. By Corollary 2 there exist ω, $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$ such that $\langle 0|W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)^\dagger Z W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(x)|0\rangle = F(x)$. Then we have

$$\mathrm{tr}[(Z^{(0)} \otimes I)\hat{\rho}] = \mathrm{tr}\left[(Z^{(0)} \otimes I)V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)\left(\sum_{j,k=0}^{2^n-1}\langle\chi_j|\rho|\chi_k\rangle|0, \chi_j\rangle\langle 0, \chi_k|\right)V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)^\dagger\right] \tag{B23}$$

$$= \sum_{j=0}^{2^n-1} p_j \mathrm{tr}[\langle 0, \chi_j | V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U)^\dagger (Z^{(0)} \otimes I) V^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(U) | 0, \chi_j\rangle] \quad (\text{where } p_j := \langle \chi_j | \rho | \chi_j \rangle) \tag{B24}$$

$$= \sum_{j=0}^{2^n-1} p_j \langle 0 | W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(\tau_j)^\dagger Z W^L_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(\tau_j) | 0 \rangle \quad (\text{by Lemma 2}) \tag{B25}$$

$$= \sum_{j=0}^{2^n-1} p_j F(\tau_j). \tag{B26}$$

■

APPENDIX C: DETAILED ANALYSIS OF QUANTUM PHASE SEARCH

1. Proof of Lemma 3

First we show that there exists a trigonometric polynomial that approximates the square wave function $\mathcal{S}(x) := \mathrm{sgn}(\sin x)$, following the results of approximating the sign function by polynomials [4].

Lemma 6 (trigonometric approximation of the square wave function). For any $\Delta, \varepsilon \in (0, 1)$ there exists a trigonometric polynomial $F \in \mathbb{C}[e^{-ix}, e^{ix}]$ of degree $L = O(\frac{1}{\Delta} \ln \frac{1}{\varepsilon})$ such that for all $x \in [-\pi, \pi]$, $|F(x)| \leqslant 1$, and for all $x \in [-\pi + \Delta, -\Delta] \cup [\Delta, \pi - \Delta]$, $|\mathcal{S}(x) - F(x)| \leqslant \varepsilon$, where $\mathcal{S}(x) := \mathrm{sgn}(\sin x)$ is the square wave function.

Proof. By Lemma 25 in [4], there exists a polynomial $P \in \mathbb{R}[x]$ of degree $L = O(\frac{1}{\delta} \ln \frac{1}{\varepsilon})$ such that $|P(x)| \leqslant 1$ for all $x \in [-1, 1]$ and $|P(x) - \mathrm{sgn}(x)| \leqslant \varepsilon$ for all $x \in [-1, 1] \setminus (-\delta, \delta)$. Letting $\delta = \sin(\Delta)$ and writing the polynomial P in a Chebyshev form as $P(x) = \sum_{j=0}^{L} a_j T_j(x)$ for some $a_j \in \mathbb{C}$, then by a change of variable

$$P(\sin x) = \sum_{j=0}^{L} \frac{(-i)^j a_j}{2} [e^{ijx} + (-1)^j e^{-ijx}] \tag{C1}$$

is a trigonometric polynomial of degree $L = O(\frac{1}{\delta} \ln \frac{1}{\varepsilon}) = O(\frac{1}{\Delta} \ln \frac{1}{\varepsilon})$. Simply letting $F(x) = P(\sin x)$, then we have $|F(x)| \leqslant 1$ for all $x \in [-\pi, \pi]$ and $|\mathrm{sgn}(\sin x) - F(x)| \leqslant \varepsilon$ for all $x \in [-\pi + \Delta, -\Delta] \cup [\Delta, \pi - \Delta]$. ■

Then we show that there exists a QPP that utilizes the approximated square wave function to classify phases on the interval $[-\pi + \Delta, -\Delta] \cup [\Delta, \pi - \Delta]$.

The following is the proof of Lemma 3.

Proof. By Lemma 6 there exists a trigonometric polynomial $f(x)$ approximating the square wave function with order $L = O(\frac{1}{\Delta} \ln \frac{1}{\varepsilon})$ such that L is a multiple of 4 and $|f(x) - \mathcal{S}(x)| < \varepsilon$ for all $x \in (-\pi + \Delta, -\Delta] \cup [\Delta, \pi - \Delta)$. It follows from Lemma 1 that there exist trigonometric polynomials $P = \sqrt{\frac{1+f(x)}{2}}$ and $Q = \sqrt{\frac{1-f(x)}{2}}$. By Theorem 1 there exist parameters $\omega \in \mathbb{R}$ and $\boldsymbol{\theta}, \boldsymbol{\phi} \in \mathbb{R}^{L+1}$ such that

$$V(U)|0, \chi_k\rangle = \sqrt{\frac{1 + f(\tau_k)}{2}} |0, \chi_k\rangle + \sqrt{\frac{1 - f(\tau_k)}{2}} |1, \chi_k\rangle. \tag{C2}$$

Define $\varepsilon_k := \frac{1}{2} |f(\tau_k) - \mathcal{S}(\tau_k)|$. For $\tau_k \in (-\pi + \Delta, -\Delta]$,

$$V(U)|0, \chi_k\rangle = \sqrt{\varepsilon_k} |0, \chi_k\rangle + \sqrt{1 - \varepsilon_k} |1, \chi_k\rangle, \tag{C3}$$

and for $\tau_k \in [\Delta, \pi - \Delta)$,

$$V(U)|0, \chi_k\rangle = \sqrt{1 - \varepsilon_k} |0, \chi_k\rangle + \sqrt{\varepsilon_k} |1, \chi_k\rangle. \tag{C4}$$

Since $|f(x) - \mathcal{S}(x)| < \varepsilon$ on $(-\pi + \Delta, -\Delta] \cup [\Delta, \pi - \Delta)$, we have $\varepsilon_k < \varepsilon$ for each $\tau_k \in (-\pi + \Delta, -\Delta] \cup [\Delta, \pi - \Delta)$. ■

2. Phase interval search

The phase interval is the region containing the eigenphase of the input state. According to Lemma 3, the main idea is to iteratively shrink the phase interval by the binary search method. At each iteration, we can decide the next subinterval, either $(-\pi + \Delta, -\Delta]$ or $[\Delta, \pi - \Delta)$, depending on the measurement result of the ancilla qubit. As Δ is small, the length of the interval is reduced by nearly half and thus the size would exponentially converge to 2Δ.

Formally, let U be a unitary, $\Delta \in (0, \frac{1}{2})$, and $\varepsilon \to 0$ (detailed analysis of success probability is discussed in Appendix C 3). Suppose $V(\cdot)$ is the quantum circuit stated in Lemma 3 with respect to Δ and ε. Define $\mathcal{G} := [-\pi, -\pi + \Delta] \cup [-\Delta, \Delta] \cup [\pi - \Delta, \pi]$ as the area that produces garbage information. Then for any quantum state $|\psi\rangle = \sum_{j=0}^{2^n-1} c_j |\chi_j\rangle$ and $\zeta \in [-\pi, \pi]$ we have

$$\begin{aligned} V(e^{-i\zeta} U)|0, \psi\rangle &= |0\rangle \left(\sum_{j: \Delta < \tau_j - \zeta < \pi - \Delta} c_j |\chi_j\rangle + \sum_{j: \tau_j - \zeta \in \mathcal{G}} g_j^{(0)} |\chi_j\rangle \right) \\ &\quad + |1\rangle \left(\sum_{j: -\pi + \Delta < \tau_j - \zeta < -\Delta} c_j |\chi_j\rangle + \sum_{j: \tau_j - \zeta \in \mathcal{G}} g_j^{(1)} |\chi_j\rangle \right), \end{aligned} \tag{C5}$$

where $g_j^{(i)}$ are garbage coefficients corresponding to state $|i\rangle$ of the ancilla qubit. Consequently, the measurement of the ancilla qubit can identify the interval containing the remaining eigenphases of the measured state. Note that the above approach is no longer applicable if the length of the interval is close to 2Δ, in which case $\tau_j - \zeta$ will always fall into the garbage area $\mathcal{G}$. Such an interval is referred to as indistinguishable. We could iteratively apply the binary search procedure to shrink the phase interval until it becomes indistinguishable. The following corollary guarantees that Algorithm 1 can reduce the length of the input interval close to 2Δ with high probability.

Lemma 7. Suppose Δ, ε, and $\mathcal{Q}$ are inputs of Algorithm 1. Then the algorithm outputs an interval $[\zeta_l, \zeta_r]$ such that $|\zeta_r - \zeta_l| = 2(\Delta + \frac{\pi}{2^{\mathcal{Q}+1}})$ and $\tau \in [\zeta_l, \zeta_r]$ with probability at least $(1-\varepsilon)^{\mathcal{Q}}$.

Proof. Define $[\zeta_l^{(j)}, \zeta_r^{(j)}]$ as the interval generated at the end of the jth iteration and $\zeta_m^{(j)}$ as the middle point of this interval. Observe that for $j > 0$ the interval generated at the end of the $(j+1)$th iteration is either $(\zeta_m^{(j)} - \Delta, \zeta_r^{(j)})$ or $(\zeta_l^{(j)}, \zeta_m^{(j)} + \Delta)$. By induction we have

$$\begin{aligned}
&\left|\zeta_r^{(\mathcal{Q})} - \zeta_l^{(\mathcal{Q})}\right| \\
&\quad = \Delta + \frac{\left|\zeta_r^{(\mathcal{Q}-1)} - \zeta_l^{(\mathcal{Q}-1)}\right|}{2} \\
&\quad = \Delta + \frac{1}{2}\left(\Delta + \frac{\left|\zeta_r^{(\mathcal{Q}-2)} - \zeta_l^{(\mathcal{Q}-2)}\right|}{2}\right) && \text{(C6)} \\
&\quad = \Delta + \frac{\Delta}{2} + \frac{\Delta}{2^2} + \cdots + \frac{\Delta}{2^{\mathcal{Q}-1}} + \frac{\left|\zeta_r^{(0)} - \zeta_l^{(0)}\right|}{2^{\mathcal{Q}}} && \text{(C7)} \\
&\quad = \Delta\left(\frac{1-2^{-\mathcal{Q}}}{1-\frac{1}{2}}\right) + \frac{2\Delta+\pi}{2^{\mathcal{Q}}} && \text{(C8)} \\
&\quad = 2\Delta + \frac{\pi}{2^{\mathcal{Q}}}, && \text{(C9)}
\end{aligned}$$

as required. By Lemma 3 the probability of failing to decide the correct subinterval is at most ε in each iteration; thus the success probability of outputting an phase interval containing τ is at least $(1-\varepsilon)^{\mathcal{Q}}$. ■

3. Phase search through interval amplification

Letting $\bar{\Delta} := \Delta + \frac{\pi}{2^{\mathcal{Q}+1}}$, merely applying Algorithm 1 will not provide an estimate within expected precision, given that the error is at most the $2\bar{\Delta}$. To address this issue, the main idea is to execute the phase interval search procedure on $(e^{i\zeta}U)^d$ for some appropriate integer d so that the binary search procedure can continue to locate the amplified phase $d\tau \in [d\zeta_l, d\zeta_r]$, since the interval length is $2d\bar{\Delta} \gg 2\Delta$. Repeating the entire procedure can exponentially reduce estimation error.

Let us formally describe the entire procedure. In the first round of phase interval search, the initial phase interval is $[-\pi, \pi]$. We iteratively apply QPP $V(\cdot)$ on the target unitary $U^{(0)} = U$ to obtain a phase interval $[\zeta_l^{(0)}, \zeta_r^{(0)}]$ of length $2\bar{\Delta}$. We define $\zeta_m^{(0)} := (\zeta_l^{(0)} + \zeta_r^{(0)})/2$ as the middle point of the interval. Letting $d := \lfloor 1/\bar{\Delta} \rfloor$, we then construct a unitary as $U^{(1)} = (e^{-i\zeta_m^{(0)}} U^{(0)})^d$ so that the interval of the amplified phase $d\tau$ is rescaled to $[-1, 1]$. Therefore, we can run the phase interval search procedure on $U^{(1)}$ to retrieve a new interval $[\zeta_l^{(1)}, \zeta_r^{(1)}]$ of length $2\bar{\Delta}$. Repeating the entire procedure above gives an estimation of phase τ up to required precision.

Now we analyze how the above procedures improve the eigenphase estimation precision. For the target eigenphase τ of the input unitary U, the corresponding phase of unitary $U^{(1)}$ is $d(\tau - \zeta_m^{(0)}) \in [\zeta_l^{(1)}, \zeta_r^{(1)}]$. Let $\zeta_m^{(1)} = (\zeta_l^{(1)} + \zeta_r^{(1)})/2$ denote the middle point of the second interval $[\zeta_l^{(1)}, \zeta_r^{(1)}]$. Then we can readily give an inequality that characterizes the estimation error,

$$\left|\zeta_m^{(1)} - d(\tau - \zeta_m^{(0)})\right| \leqslant \bar{\Delta}. \tag{C10}$$

We further rewrite this inequality as

$$\left|\tau - \left(\zeta_m^{(0)} + \frac{\zeta_m^{(1)}}{d}\right)\right| \leqslant \frac{\bar{\Delta}}{d} \leqslant d^{-2}. \tag{C11}$$

From this equation we can see that this scheme gives an estimate of eigenphase with an error of d^{-2}. After repeating the procedure sufficiently many times, we inductively obtain a sequence $(\zeta_m^{(0)}, \zeta_m^{(1)}, \ldots)$ that could be taken as an estimate of the eigenphase. Therefore, the estimation error will exponentially decay with iterating. For instance, assuming our scheme is executed $\mathcal{T}$ times, it will inductively give an estimate as

$$\left|\tau - \left(\zeta_m^{(0)} + \frac{\zeta_m^{(1)}}{d} + \frac{\zeta_m^{(2)}}{d^2} + \cdots + \frac{\zeta_m^{(\mathcal{T}-1)}}{d^{\mathcal{T}-1}}\right)\right| \leqslant \frac{\bar{\Delta}}{d^{\mathcal{T}-1}} \leqslant d^{-\mathcal{T}}. \tag{C12}$$

Note that $d = [1/\bar{\Delta}]$ must be at least 2; otherwise we cannot find a sequence that converges to the eigenphase. Also, if the estimation precision is expected to be δ, then $\mathcal{T}$ should satisfy $d^{-\mathcal{T}} \leqslant \delta$. Then we derive that

$$\mathcal{Q} = \left\lceil \ln\left(\frac{2\pi}{1-2\Delta}\right)\right\rceil, \quad \mathcal{T} = \left\lceil \frac{\ln(\frac{1}{\delta})}{\ln(d)} \right\rceil. \tag{C13}$$

The entire phase search procedure executes the phase interval search, i.e., Algorithm 1, $\mathcal{T}$ times. Then by Lemma 7 the success probability of the entire phase search procedure is $(1-\eta)^{\mathcal{Q}\mathcal{T}}$, where η is the input threshold of Algorithm 1. For any $\varepsilon \in (0, 1)$, let $\eta = \frac{\varepsilon}{\mathcal{Q}\mathcal{T}}$. Then the total success probability is

$$(1-\eta)^{\mathcal{Q}\mathcal{T}} > 1 - \mathcal{Q}\mathcal{T}\eta = 1 - \varepsilon, \tag{C14}$$

where the first strict inequality is the Bernoulli inequality for $\eta < 1$ and $\mathcal{Q}\mathcal{T} \geqslant 2$. By Lemma 3, to ensure that the success probability of the entire algorithm is at least $1-\varepsilon$, the number of QPP layers L should be

$$L = O\left(\frac{1}{\Delta}\ln\frac{\mathcal{Q}\mathcal{T}}{\varepsilon}\right) = O\left(\frac{1}{\Delta}\ln\frac{\ln\frac{1}{\delta}}{\varepsilon} + g(\Delta)\right). \tag{C15}$$

Here $g(\Delta)$ is a function in terms of Δ only and hence can be omitted in the complexity analysis of Theorem 3.

Overall, by running the scheme many times, we can find an estimate of the eigenphase with precision δ and success probability at least $1-\varepsilon$. The above results are summarized in Algorithm 2.

4. Proof of Theorem 3

The following is the proof of Theorem 3.

Proof. We analyze the number of queries to the controlled-U oracle in Algorithm 2 to get an estimate within required precision $\delta > 0$, probability of failure $\varepsilon > 0$, and input Δ. Note that Δ is a self-adjusted parameter and hence can be considered as a constant in complexity analysis.

Observe that Algorithm 2 executes the phase interval search procedure $\mathcal{T}$ times, while the tth phase interval search procedure requires $\mathcal{Q}$ calls of circuit $V(U^{d^t})$ of L layers.

By Eqs. (C13) and (C15) the total query complexity of the controlled-U oracle is

$$\sum_{t=0}^{\mathcal{T}-1} \mathcal{Q} \times d^t \times L = O(d^{\mathcal{T}} L \mathcal{Q})$$
$$= O\left(\frac{1}{\delta}\frac{1}{\Delta} \ln\left(\frac{\ln\frac{1}{\delta}}{\varepsilon}\right) \ln\left(\frac{2\pi}{1-2\Delta}\right)\right)$$
$$= O\left(\frac{1}{\delta} \ln\left(\frac{1}{\varepsilon} \ln\frac{1}{\delta}\right)\right). \qquad \text{(C16)}$$

■

5. Application: Period finding and factoring

In this section, we consider applying the quantum phase search algorithm to solve the period-finding problem. The goal of period finding is to find the smallest integer r (namely, the order) of a given element x in the rings of integers modulo $N \in \mathbb{N}$ such that $x^r \equiv 1$ (mod$N$).

In the quantum setting, the problem of a reversible quantum modular multiplier has been well studied [60,61]. It has been proved that the quantum operator

$$U_x|y \,\mathrm{mod} N\rangle = |xy \,\mathrm{mod} N\rangle, \quad y \in \mathbb{Z}/N\mathbb{Z}, \qquad \text{(C17)}$$

can be constructed in cubic resources for every integer $x \in \mathbb{Z}/N\mathbb{Z}$. A novel property of such modular multiplier is that there is no additional quantum cost to realize a power of U_x since $U_x^j = U_{x^j}$. Moreover, an eigenvector of U_x is in the form

$$|u_s\rangle := \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} \exp\left(\frac{2\pi isk}{r}\right) |x^k \mathrm{mod} N\rangle, \qquad \text{(C18)}$$

corresponding to its eigenphase $\frac{2\pi s}{r}$, and the uniform superposition of all eigenvectors is $|1\rangle$, i.e., $|1\rangle = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle$.

The conventional quantum period-finding algorithm is to apply the quantum phase estimation algorithm to the modular multiplier with input state $|1\rangle$ and then use the continued fraction algorithm to extract the order from the estimated phases. Similarly, we use the quantum phase search algorithm to extract eigenphases of the modular operator. The details of the algorithm are shown below.

Algorithm 4. Quantum period finding.

```
Require: $N \in \mathbb{Z}, x \in \mathbb{Z}/N\mathbb{Z}$, constant $\Delta \in (0, \frac{1}{2})$, and error
    tolerance $\varepsilon, \delta > 0$.
Ensure: order $r$ of $x$ in $\mathbb{Z}/N\mathbb{Z}$.
Construct the modular multiplier $U_x$ by $x$ and $N$.
Retrieve an estimated eigenphase $\tau$ of $U_x$ by the QPS
    algorithm, that is, $\tau \leftarrow \mathrm{QPS}(U_x, |1\rangle, \Delta, \varepsilon, \delta)$.
Apply a continued-fractional algorithm on $\tau$ to retrieve $l$ and $r$;
    return $r$.
```

6. Application: Amplitude estimation

The problem of quantum amplitude estimation can be efficiently solved by the phase estimation algorithm [21], providing a quadratic speedup over classical Monte Carlo methods. In recent years, several studies [35,62–64] have realized phase-estimation-free amplitude estimation with the same quantum speedup. However, these works require a large number of samplings, i.e., measurements from quantum circuits. Here we show that our phase search algorithm can also apply to the amplitude estimation and inherit the computational advantage from conventional phase estimation.

Let $\mathcal{A}$ denote a quantum circuit that acts on n qubits. Applying the circuit $\mathcal{A}$ to $|0^n\rangle$, the produced state is of the form

$$\mathcal{A}|0^{\otimes n}\rangle = \cos(\tau)|0\rangle|\psi\rangle + \sin(\tau)|1\rangle|\phi\rangle, \qquad \text{(C19)}$$

where $|\psi\rangle$ and $|\phi\rangle$ are $(n-1)$-qubit states and $\tau \in (-\pi, \pi)$ is the phase. Here $\cos(\tau)$ and $\sin(\tau)$ denote the amplitudes of states $|\psi\rangle$ and $|\phi\rangle$, respectively. Our goal is to estimate $|\sin(\tau)|$ up to a given precision with high probability.

Suppose we can repeatedly apply the circuit $\mathcal{A}$ and its inverse $\mathcal{A}^\dagger$. Then we can construct a circuit for the Grover operator

$$G = \mathcal{A}(2|0^{\otimes n}\rangle\langle 0^{\otimes n}| - I^{\otimes n})\mathcal{A}^\dagger(I - 2|1\rangle\langle 1|) \otimes I^{\otimes(n-1)}. \qquad \text{(C20)}$$

Note that, in the space spanned by $\{|0\rangle|\psi\rangle, |1\rangle|\phi\rangle\}$, the Grover operator G has an eigenphase 2τ or -2τ. Thus our amplitude estimation algorithm just applies the quantum phase search algorithm to the Grover operator. Specifically, applying our quantum phase search algorithm can effectively extract eigenphases $\pm 2\tau$. Moreover, postprocessing can estimate the amplitude within the required precision. We show more details below.

Algorithm 5. Quantum amplitude estimation.

```
Require: circuit $\mathcal{A}$, constant $\Delta \in (0, \frac{1}{2})$, and error tolerance
    $\varepsilon, \delta > 0$.
Ensure: an amplitude $\sin(\tau)$.
Prepare the initial state $|\chi\rangle = \mathcal{A}|0\rangle^{\otimes n}$ and construct the Grover
    operator $G$ in Eq. (C20).
Retrieve an estimated eigenphase $\tilde{\tau}$ of $G$ by the QPS algorithm,
    that is, $\tilde{\tau} \leftarrow \mathrm{QPS}(G, |\chi\rangle, \Delta, \varepsilon, \delta)$.
Return $|\sin(\frac{\tilde{\tau}}{2})|$.
```

APPENDIX D: FURTHER REVIEW FOR BLOCK ENCODING

1. Qubitization

In this section we review the technique of qubitization purposed by [7]. Such a technique associates the spectrum of target block encoding U_A with the block encoded matrix A. Assume A is Hermitian with $\|A\| \leqslant 1$, since our work only deal with Hamiltonians and density operators. Recent work has discussed an explicit construction scheme for building a block encoding sparse matrix [65]. To better understand qubitization, we analyze the spectral information of the circuit in Fig. 4. Let $\hat{U}_A$ denote the qubitization of block encoding and U_A and $\tilde{U}_A$ denote the unitaries for the dashed region in Fig. 4. Then Lemma 10 of [7] implies that $\tilde{U}_A$ satisfies

$$(\langle 0^{\otimes(m+1)}| \otimes I^{\otimes n})\, \tilde{U}_A\, (|0^{\otimes(m+1)}\rangle \otimes I^{\otimes n}) = A, \qquad \text{(D1)}$$

$$(\langle 0^{\otimes(m+1)}| \otimes I^{\otimes n})\, \tilde{U}_A^2\, (|0^{\otimes(m+1)}\rangle \otimes I^{\otimes n}) = I^{\otimes n}. \qquad \text{(D2)}$$

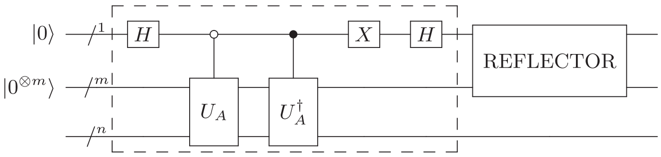


FIG. 4. Circuit realization for the qubitized unitary $\hat{U}_A$. Here U_A is an $(n+m)$-qubit block encoding of an n-qubit matrix A and the REFLECTOR gate is equivalent to $2|0^{\otimes(m+1)}\rangle\langle 0^{\otimes(m+1)}| - I^{\otimes(m+1)}$. Note that it suffices to control the REFLECTOR and three X gates to realize a controlled-$\hat{U}_A$ gate.

Let $|\psi_\lambda\rangle$ be an eigenstate of A corresponding to its eigenvalue λ. Define $|\hat{\psi}_\lambda\rangle := |0^{\otimes(m+1)}, \psi_\lambda\rangle$. After respectively applying $\tilde{U}_A$ and $\tilde{U}_A^2$ to $|\hat{\psi}_\lambda\rangle$, states are of the form

$$\tilde{U}_A|\hat{\psi}_\lambda\rangle = \lambda|\hat{\psi}_\lambda\rangle + \sqrt{1-\lambda^2}|\hat{\perp}_\lambda\rangle, \tag{D3}$$

$$\tilde{U}_A^2|\hat{\psi}_\lambda\rangle = |\hat{\psi}_\lambda\rangle, \tag{D4}$$

where $|\hat{\perp}_\lambda\rangle$ is an orthogonal state that satisfies $(|0^{\otimes(m+1)}\rangle\langle 0^{\otimes(m+1)}| \otimes I^{\otimes n})|\hat{\perp}_\lambda\rangle = 0$. The above results also imply that all subspaces $\text{span}\{|\hat{\psi}_\lambda\rangle, |\hat{\perp}_\lambda\rangle\}$ are mutually perpendicular. Moreover, Eq. (D4) implies

$$\tilde{U}_A|\hat{\perp}_\lambda\rangle = \sqrt{1-\lambda^2}|\hat{\psi}_\lambda\rangle - \lambda|\hat{\perp}_\lambda\rangle. \tag{D5}$$

Note that it suffices to analyze $\hat{U}_A$ under subspace

$$\mathcal{H}_A := \bigoplus_\lambda \text{span}\{|\hat{\psi}_\lambda\rangle, |\hat{\perp}_\lambda\rangle\}, \tag{D6}$$

since the input state of ancilla qubits is always $|0^{\otimes(m+1)}\rangle$. In this subspace, we can see that $\hat{U}_A$ is essentially a rotation whose matrix is similar to R_Y, i.e.,

$$\begin{aligned}\hat{U}_A &= (\text{REFLECTOR} \otimes I^{\otimes n})\tilde{U}_A \\ &= \bigoplus_\lambda \begin{bmatrix} \lambda & -\sqrt{1-\lambda^2} \\ \sqrt{1-\lambda^2} & \lambda \end{bmatrix}_{\{|\hat{\psi}_\lambda\rangle, |\hat{\perp}_\lambda\rangle\}} \oplus [\cdots]_{\mathcal{H}_A^\perp}.\end{aligned} \tag{D7}$$

The spectral details of $\hat{U}_A$ in $\mathcal{H}$ follow immediately:

$$\begin{aligned}&\text{eigenvector } |\chi_\lambda^+\rangle = \frac{1}{\sqrt{2}}(|\hat{\psi}_\lambda\rangle + i|\hat{\perp}_\lambda\rangle), \quad \text{eigenvalue } e^{+i\tau_\lambda}, \quad \text{where } \lambda = \cos(\tau_\lambda);\\ &\text{eigenvector } |\chi_\lambda^-\rangle = \frac{1}{\sqrt{2}}(|\hat{\psi}_\lambda\rangle - i|\hat{\perp}_\lambda\rangle), \quad \text{eigenvalue } e^{-i\tau_\lambda}, \quad \text{where } \lambda = \cos(-\tau_\lambda)\end{aligned} \tag{D8}$$

Therefore, we can select $\hat{U}_A$ as the input unitary in the QPP circuit, to access the arc cosine of eigenvalues of A, allowing phase evolution and evaluation to be applied on block encoded matrices.

2. Block encoding construction for density matrices

The quantum purification model, which prepares a purification of a mixed state ρ, is an extensively explored model for entropy in the literature. Consider quantum registers A and B storing n and n' qubits, respectively. Suppose we have access to an $(n+n')$-qubit unitary oracle U_ρ acting on these two registers such that

$$|\Psi\rangle_{AB} := U_\rho|0^{\otimes n}\rangle_A|0^{\otimes n'}\rangle_B, \quad \text{tr}_B(|\Psi\rangle_{AB}\langle\Psi|_{AB}) = \rho. \tag{D9}$$

Such an oracle can be further employed to construct a block encoding $\hat{U}_\rho$ of ρ. We recall Lemma 7 in [7] and give the circuit construction for $\hat{U}_\rho$ in Fig. 5, using U_ρ and $U_\rho^\dagger$ once.

Note that the unitary $\tilde{U}_\rho$ for the dashed region in Fig. 5 satisfies

$$(\langle 0^{\otimes(n+n')}|_{AB} \otimes I^{\otimes n})\,\tilde{U}_\rho\,(|0^{\otimes(n+n')}\rangle_{AB} \otimes I^{\otimes n}) = \rho, \tag{D10}$$

$$(\langle 0^{\otimes(n+n')}|_{AB} \otimes I^{\otimes n})\,\tilde{U}_\rho^2\,(|0^{\otimes(n+n')}\rangle_{AB} \otimes I^{\otimes n}) = I^{\otimes n}. \tag{D11}$$

Define $\{p_j\}_{j=0}^{2^n-1}$ as the set of eigenvalues of ρ and $\{\tau_j\}_{j=0}^{2^{n+1}-1}$ as the set of eigenphases of $\hat{U}_\rho$ under subspace $\mathcal{H}_\rho$ defined in Eq. (D6). Using the same reasoning as in Appendix D 1, we have

$$\{\tau_j\}_{j=0}^{2^{n+1}-1} = \{\arccos(p_j), -\arccos(p_j)\}_{j=0}^{2^n-1}. \tag{D12}$$

As shown above, the spectrum of $\hat{U}_\rho$ is connected to that of ρ in subspace $\mathcal{H}_\rho$.

APPENDIX E: FURTHER DISCUSSION OF HAMILTONIAN PROBLEMS

1. Hamiltonian simulation

In this section, we explain the main idea of our method for Hamiltonian simulation and discuss how to find parameters for simulating the target function $f(x)$. For convenience, let $\hat{U}_H$ denote the qubitized block encoding of a Hamiltonian H.

Prepare the initial state $|0^{\otimes(m+2)}, \psi\rangle$, where the first qubit is the ancilla qubit of QPP and the other $m+1$ qubits are ancilla

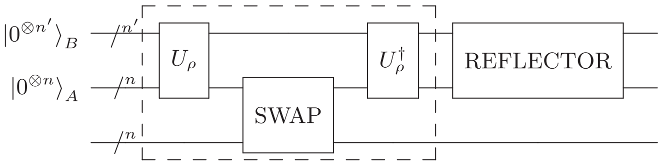


FIG. 5. Circuit realization for $\hat{U}_\rho$. The input state for two ancilla registers is $|0^{\otimes(n+n')}\rangle_{AB}$. Here SWAP is a $2n$-qubit SWAP operator and REFLECTOR is equivalent to $2|0^{\otimes(n+n')}\rangle\langle 0^{\otimes(n+n')}| - I^{\otimes(n+n')}$. Note that it suffices to control the SWAP and REFLECTOR gates to realize a controlled-$\hat{U}_\rho$ unitary.

qubits of the qubitized block encoding $\hat{U}_H$. Decompose the initial state $|0^{\otimes(m+2)}, \psi\rangle$ by eigenvectors of the Hamiltonian

$$|0\rangle|0^{\otimes(m+1)}, \psi\rangle = \sum_\lambda \beta_\lambda |0\rangle|0^{\otimes(m+1)}, \psi_\lambda\rangle, \tag{E1}$$

where $\sum_\lambda |\beta_\lambda|^2 = 1$. As shown in Eq. (D7), the qubitized block encoding $\hat{U}_H$ is a rotation in each subspace $\text{span}\{|0^{\otimes(m+1)}, \psi_\lambda\rangle, |\hat{\perp}_\lambda\rangle\}$. Then we construct the circuit of Hamiltonian simulation by incorporating $\hat{U}_H$ into the structure of QPP. Note that one eigenvalue of the Hamiltonian corresponds to two eigenvalues of $\hat{U}_H$ and then the action of the circuit can be described as the matrix below, with respect to the basis of $\{|0, \chi_\lambda^+\rangle, |1, \chi_\lambda^+\rangle\}$ and $\{|0, \chi_\lambda^-\rangle, |1, \chi_\lambda^-\rangle\}$ for each eigenvalue λ,

$$\bigoplus_\lambda \left(\begin{bmatrix} P(\tau_\lambda) & -Q(\tau_\lambda) \\ Q^*(\tau_\lambda) & P^*(\tau_\lambda) \end{bmatrix}_\lambda \oplus \begin{bmatrix} P(-\tau_\lambda) & -Q(-\tau_\lambda) \\ Q^*(-\tau_\lambda) & P^*(-\tau_\lambda) \end{bmatrix}_\lambda \right). \tag{E2}$$

On the other hand, note that each state $|0^{m+1}, \psi_\lambda\rangle$ can be written as an equal-weighted sum of eigenvectors of $\hat{U}_H$, implying

$$|0^{\otimes(m+2)}, \psi_\lambda\rangle = \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}|0\rangle(|0^{\otimes(m+1)}, \psi_\lambda\rangle + i|\hat{\perp}_\lambda\rangle) + \frac{1}{\sqrt{2}}|0\rangle(|0^{\otimes(m+1)}, \psi_\lambda\rangle - i|\hat{\perp}_\lambda\rangle) \right). \tag{E3}$$

Using Eq. (E3), we thus rewrite the initial state as a superposition of eigenvectors of $\hat{U}_H$. With the decomposition in Eq. (E2), applying QPP to the state $|0^{m+2}, \psi\rangle$ outputs a state of the form

$$\sum_\lambda \frac{\beta_\lambda}{\sqrt{2}} \left(\frac{(P(\tau_\lambda)|0\rangle + Q^*(\tau_\lambda)|1\rangle)}{\sqrt{2}}(|0^{\otimes(m+1)}, \psi_\lambda\rangle + i|\hat{\perp}_\lambda\rangle) + \frac{(P(-\tau_\lambda)|0\rangle + Q^*(-\tau_\lambda)|1\rangle)}{\sqrt{2}}(|0^{\otimes(m+1)}, \psi_\lambda\rangle - i|\hat{\perp}_\lambda\rangle) \right). \tag{E4}$$

The output state is near the target state as much as possible by suitably truncating the target function. In fact, the difference between the final output state and the target state is bounded by the quantity

$$\epsilon \leqslant \max_{x\in[-\pi,\pi]} \|[P(x)|0\rangle + Q^*(x)|1\rangle] - e^{-i\cos(x)t}|0\rangle\| \tag{E5}$$

$$= \max_{x\in[-\pi,\pi]} \sqrt{|P(x) - e^{-i\cos(x)t}|^2 + 1 - |P(x)|^2}, \tag{E6}$$

where ϵ denotes error. Let $a_x = P(x) - e^{-i\cos(x)t}$ denote the difference and assume that $|a_x| \leqslant \varepsilon$ for all $x \in [-\pi, \pi]$. Then we show how large $|P(x)|^2$ is:

$$|a_x|^2 = [P(x) - e^{-i\cos(x)t}][P^*(x) - e^{i\cos(x)t}] \Rightarrow |P(x)|^2$$

$$= 2\,\text{Re}[e^{i\cos(x)t}P(x)] + |a_x|^2 - 1 \tag{E7}$$

$$\geqslant 1 - 2|a_x| + |a_x|^2 = (1 - |a_x|)^2. \tag{E8}$$

The state approximation error is at most as large as

$$\epsilon \leqslant \max_{x\in[-\pi,\pi]} \sqrt{|a_x|^2 + 1 - (1 - |a_x|)^2} \leqslant \sqrt{2\varepsilon}. \tag{E9}$$

Hence, Eq. (E9) establishes the relation between the state approximation and the function approximation.

It remains to show $P(x)$ can approximate $f(x) = e^{-i\cos(x)t}$ with arbitrary precision, which is true since $f(x)$ could be expanded into a trigonometric polynomial. We summarized the results in Theorem 9 and discuss how to truncate $f(x)$ in the proof. The following is the proof of Theorem 9.

Proof. By Theorem 1, QPP can simulate any trigonometric polynomial with an order N. Thus we just need to find such a polynomial that approximates the function $f(x) = e^{-i\cos(x)t}$ within precision $\delta^2/2$. We recall the Jacobi-Anger expansion $e^{iz\cos(\theta)} = \sum_{k=-\infty}^{\infty} i^k J_k(z) e^{ik\theta}$ [66], where $J_k(z)$ is the kth Bessel function of the first kind. The truncation error of the Jacobi-Anger expansion has been well studied in the literature [4,67]. Given the truncation error $\delta^2/2$, the order N is given by

$$N = \Theta\left(|z| + \frac{\ln(2/\delta^2)}{\ln\left(e + \frac{\ln(2/\delta^2)}{|z|}\right)} \right). \tag{E10}$$

Then QPP circuit uses N times controlled U_H and $U_H^\dagger$. Recall that the evolution time is $t\Lambda$; thus we set the parameter $z = t\Lambda$. Clearly, the cost of the circuit is the same as claimed, and the proof is finished. ■

2. Hamiltonian eigensolver

A fundamental problem in physics is to calculate static properties of a quantum system. Of all the questions one might ask about a quantum system, the most frequently asked is what the energy and eigenstates of a Hamiltonian are. After the development of quantum phase estimation, numerous quantum algorithms for computing the spectrum of a Hamiltonian have been developed. Clearly, we could just directly apply our phase search algorithm in a similar spirit to give quantum algorithms for extracting Hamiltonian eigeninformation. To this end, we describe several quantum algorithms based on quantum phase estimation below.

The quantum algorithm proposed by Abrams and Lloyd [68] computes the eigenstates and eigenvalues of a Hamiltonian via applying quantum phase estimation to the simulated real-time evolution operator, resulting in exponential speedups over known classical methods. Under postselection of the ancilla register, this approach can output the spectrum by reading an eigenvalue and obtaining the corresponding eigenvector. Despite being conceptually simple, the technical realization of the synthesis between quantum phase estimation and Hamiltonian simulation could be difficult. Indeed, realizing the evolution operator requires a potentially large circuit, which would complicate the entire phase estimation circuit and make its practical implementation challenging.

In fact, the use of the time-evolution operator is not necessary for phase estimation. Work by Poulin *et al.* [69] proposed a method that extracts the spectrum of a Hamiltonian H from a qubitized block encoding of H/Λ. The idea is to estimate a phase of the qubitized block encoding $\hat{U}$ and then directly calculate the corresponding eigenvalue of H by the relationship between their spectrum, i.e., $\lambda = \Lambda\cos(\tau_\lambda)$. To achieve the desired precision δ, one needs to apply the block encoding $O(\Lambda/\delta)$ times.

Moreover, the eigenvector of the Hamiltonian can be extracted from the postmeasurement state of quantum phase estimation. Specifically, suppose we obtain an eigenvector $|\chi_\lambda^\pm\rangle = \frac{1}{\sqrt{2}}(|0^{\otimes(m+1)}, \psi_\lambda\rangle \pm i|\perp_\lambda\rangle)$ by phase estimation. Since $(|0^{\otimes(m+1)}\rangle\langle 0^{\otimes(m+1)}| \otimes I^{\otimes n})|\perp_\lambda\rangle = 0$, we directly measure the ancilla register of the qubitized block encoding in the computational basis. Clearly, the probability of obtaining all zeros from the measurement is exactly half, and the postmeasurement state is the corresponding eigenvector. If the result is not as expected, we could reapply the phase estimation and repeat the measurement, projecting the state into $|\chi_\lambda^+\rangle$ or $|\chi_\lambda^-\rangle$. Repeating k times makes the failure probability decay exponentially as $(1/2)^k$.

Another important Hamiltonian problem is to compute ground- and excited-state properties of many-body systems, which is quite a challenge in quantum physics and quantum chemistry. Recently, Dong *et al.* [17] proposed a method via quantum eigenvalue transformation of unitary matrices, which could be naturally generalized into the structure of QPP.

APPENDIX F: THEOREMS OF ENTROPY ESTIMATION

1. Proof of Theorem 4

The following is the proof of Theorem 4.

Proof. We start with the spectral decomposition of σ and $\hat{U}_\sigma$ under subspace $\mathcal{H}_\sigma$ defined in Eq. (D6). From Eq. (D8) we have

$$\sigma = \sum_{j=0}^{2^n-1} q_j|\psi_j\rangle\langle\psi_j|, \tag{F1}$$

$$\hat{U}_\sigma = \bigoplus_{j=0}^{2^n-1} \begin{bmatrix} e^{i\tau_j} & 0 \\ 0 & e^{-i\tau_j} \end{bmatrix}_{\{|\chi_j^+\rangle, |\chi_j^-\rangle\}} \oplus [\cdots]_{\mathcal{H}_\sigma^\perp}, \tag{F2}$$

where $\tau_j = \arccos(q_j)$ and $|\chi_j^\pm\rangle$ are eigenstates of $\hat{U}_\sigma$ such that

$$\begin{aligned} |\chi_j^+\rangle &= \frac{1}{\sqrt{2}}(|0^{\otimes m}, \psi_j\rangle + i|\hat{\perp}_j\rangle), \\ |\chi_j^-\rangle &= \frac{1}{\sqrt{2}}(|0^{\otimes m}, \psi_j\rangle - i|\hat{\perp}_j\rangle) \end{aligned} \tag{F3}$$

for some quantum state $|\hat{\perp}_j\rangle$ (defined in Appendix D 1) so that $(|0^{\otimes m}\rangle\langle 0^{\otimes m}| \otimes I^{\otimes n})|\hat{\perp}_j\rangle = 0$. Also, note that $|0^{\otimes m}\rangle\langle 0^{\otimes m}| \otimes \rho$ is a density matrix in $\mathcal{H}_\sigma$ and hence can be decomposed by the basis $\{|\chi_j^\pm\rangle\}_{j=0}^{2^n-1}$. Now we are ready to analyze the effect of QPP on the input state. Define the function $F(x) = \sum_{k=-L}^{L} d_k e^{ikx} := f(\cos(x))$. Note that $|F|(x) \leqslant 1$ implies $\|\boldsymbol{d}\|_1 \leqslant 1$.

Suppose the input state is a purified state $|\Psi_\rho\rangle_{AB}$ so that $\mathrm{tr}_B(|\Psi_\rho\rangle_{AB}) = \rho$. Here the register A is the main register of the QPP circuit. By Schmidt decomposition, there exists an orthonormal set $\{|\phi_j\rangle\}_{j=0}^{2^n-1}$ of quantum states on register B such that

$$|\Psi_\rho\rangle_{AB} = \sum_{j=0}^{2^n-1} \sqrt{p_j}|\psi_j\rangle_A|\phi_j\rangle_B. \tag{F4}$$

From Theorem 2 there exists a QPP circuit $V(\hat{U}_\sigma \otimes I_B)$ of L layers such that

$$\langle Z^{(0)}\rangle_{|\Phi\rangle} = \mathrm{tr}[(Z^{(0)} \otimes I^{\otimes(m+n)} \otimes I_B)|\Phi\rangle\langle\Phi|] \tag{F5}$$

$$= \sum_{j=0}^{2^n-1} p_j^+ F(\tau_j) + \sum_{j=0}^{2^n-1} p_j^- F(-\tau_j) = \sum_{j=0}^{2^n-1} (p_j^+ + p_j^-) f(q_j), \tag{F6}$$

where $p_j^\pm = \langle\chi_j^\pm|\,(|0^{\otimes m}\rangle\langle 0^{\otimes m}| \otimes \rho)\,|\chi_j^\pm\rangle$. Further, Eq. (F3) implies

$$\begin{aligned} p_j^+ + p_j^- &= \langle\chi_j^+|(|0^{\otimes m}\rangle\langle 0^{\otimes m}| \otimes \rho)|\chi_j^+\rangle \\ &\quad + \langle\chi_j^-|(|0^{\otimes m}\rangle\langle 0^{\otimes m}| \otimes \rho)|\chi_j^-\rangle \quad \text{(F7)} \\ &= \langle\psi_j|\rho|\psi_j\rangle. \quad \text{(F8)} \end{aligned}$$

The statement holds as

$$\langle Z^{(0)}\rangle_{|\Phi\rangle} = \sum_{j=0}^{2^n-1} \langle\psi_j|\rho|\psi_j\rangle f(q_j) = \mathrm{tr}[\rho f(\sigma)]. \tag{F9}$$

■

Specifically, for any entropic function that is well approximated by a polynomial, we can use QPP circuits to estimate such an entropy by Theorem 4. The following result guarantees that the number of circuit layers L for polynomial approximation is $O(\ln\frac{1}{\varepsilon})$ up to precision ε.

Lemma 8 (Corollary 66 in [4]). Let $x_0 \in [-1, 1]$, $r \in (0, 2]$, and $\delta \in (0, r]$; also let $f : [-x_0 - r - \delta, x_0 + r + \delta] \to \mathbb{C}$ and be such that $f(x_0 + x) = \sum_{k=0}^{\infty} c_k x^k$ for all $x \in [-r - \delta, r + \delta]$. Suppose $B > 0$ is such that $\sum_{k=0}^{\infty}(r + \delta)^k|c_k| \leqslant B$. Letting $\varepsilon \in (0, \frac{1}{2B}]$, then there is an efficiently computable polynomial $P \in \mathbb{C}[x]$ of degree $O(\frac{1}{\delta}\ln\frac{B}{\varepsilon})$ such that for all $x \in [-1, 1]$, $|P(x)| \leqslant \varepsilon + B$, and for all $x \in [x_0 - r, x_0 + r]$, $|f(x) - P(x)| \leqslant \varepsilon$.

2. Proofs for von Neumann and relative entropy estimation

The following is the proof of Theorem 5.

Proof. Define $f(x) = \frac{\ln(x)}{2\ln(\gamma)}$. We expect to find a polynomial $P(x) \in \mathbb{R}[x]$ such that for all $x \in [\gamma, 1]$,

$$|P(x) - f(x)| \leqslant \frac{\varepsilon}{4\ln(1/\gamma)}. \tag{F10}$$

By taking $x_0 = 1$, $r = 1 - \gamma$, $\delta = \frac{\gamma}{2}$, $B = \frac{1}{2}$, and $\eta = \frac{\varepsilon}{4\ln(1/\gamma)}$ in Lemma 8, such a polynomial P exists with degree

$$L = O\left(\frac{1}{\delta}\ln\frac{B}{\eta}\right) = O\left(\frac{1}{\gamma}\ln\frac{\ln(1/\gamma)}{\varepsilon}\right). \tag{F11}$$

Note that Lemma 11 of Ref. [4] used the same setup for Lemma 8. Then Theorem 4 implies that there exists a QPP circuit $V(\hat{U}_\rho)$ of L layers to estimate $\mathrm{tr}[\rho P(\rho)]$. Up to precision $\frac{\varepsilon}{4\ln(1/\gamma)}$, the experimental estimation E can be retrieved by measuring the first qubit of $V_{\omega,\boldsymbol{\theta},\boldsymbol{\phi}}(\hat{U}_\rho)$ with input state $|0^{\otimes(m+1)}\rangle\langle 0^{\otimes(m+1)}| \otimes \rho$. We have

$$|E - \mathrm{tr}[\rho f(\rho)]| \leqslant |E - \mathrm{tr}[\rho P(\rho)]| + |\mathrm{tr}[\rho P(\rho)] - \mathrm{tr}[\rho f(\rho)]| \tag{F12}$$

$$\leqslant \frac{\varepsilon}{4\ln(1/\gamma)} + \|P - f\|_{[\gamma,1]} \leqslant \frac{\varepsilon}{2\ln(1/\gamma)} \tag{F13}$$

and hence

$$|2\ln(\gamma)E - S(\rho)| \leqslant \varepsilon. \tag{F14}$$

To obtain the desired precision, by Chebyshev's inequality, the total number of measurements is $O(\frac{\ln^2(\gamma)}{\varepsilon^2})$, while each $\hat{U}_\rho$ requires one call of U_ρ and its inverse. Consequently, the total query complexity of U_ρ and its inverse is

$$O\left(\frac{1}{\gamma\varepsilon^2}\ln^2\left(\frac{1}{\gamma}\right)\ln\frac{\ln(1/\gamma)}{\varepsilon}\right). \tag{F15}$$

The statement for using amplitude estimation follows by switching Chebyshev's inequality to the complexity of amplitude estimation [21,35]. ■

Note that by replacing $\hat{U}_\rho$ with $\hat{U}_\sigma$, the proof of Theorem 6 is exactly the same as the proof of Theorem 5.

The following is the proof of Corollary 3.

Proof. We follow the same proof as for Theorem 5, with extra consideration of the threshold value $\gamma > 0$ and the error generated by this threshold value. We choose the same function f and a polynomial P of degree $O(\frac{1}{\gamma}\ln\frac{1}{\eta})$ such that for all $x \in [\gamma, 1]$,

$$|P(x) - f(x)| \leqslant \eta, \tag{F16}$$

where $\eta > 0$ is decided later. Denote by E the experiment estimation. Then

$$\begin{aligned} &|E - \mathrm{tr}[\rho f(\rho)]| \\ &\quad \leqslant |E - \mathrm{tr}[\rho P(\rho)]| + |\mathrm{tr}[\rho P(\rho)] - \mathrm{tr}[\rho f(\rho)]| && \text{(F17)} \\ &\quad \leqslant |E - \mathrm{tr}[\rho P(\rho)]| + \kappa\|xP - xf\|_{[0,\gamma]} + \|P - f\|_{[\gamma,1]} && \text{(F18)} \\ &\quad \leqslant |E - \mathrm{tr}[\rho P(\rho)]| + 2\kappa\gamma + \eta. && \text{(F19)} \end{aligned}$$

Choose $|E - \mathrm{tr}[\rho P(\rho)]| \leqslant \frac{\varepsilon}{8\ln(1/\gamma)}$, $\gamma = \frac{\varepsilon}{16\kappa\ln(\kappa/\varepsilon)}$, and $\eta = \frac{\varepsilon}{8\ln(1/\gamma)}$. We have

$$|2\ln(\gamma)E - S(\rho)| \leqslant \frac{\varepsilon}{4} + \frac{\varepsilon}{2} + \frac{\varepsilon}{4} = \varepsilon. \tag{F20}$$

Then the total complexity is

$$\begin{aligned} &O\left(\frac{1}{\varepsilon^2}\ln^2\left(\frac{\kappa}{\varepsilon}\right)\right)O\left(\frac{\kappa}{\varepsilon}\ln\left(\frac{\kappa}{\varepsilon}\right)\ln\left(\frac{1}{\varepsilon}\right)\right) \\ &\quad = O\left(\frac{\kappa}{\varepsilon^3}\ln^3\left(\frac{\kappa}{\varepsilon}\right)\ln\left(\frac{1}{\varepsilon}\right)\right), && \text{(F21)} \end{aligned}$$

as required. The statement for using amplitude estimation follows by switching Chebyshev's inequality to the complexity of amplitude estimation [21,35]. ■

3. Proofs for quantum Rényi entropy estimation

An extra lemma is required to proceed to the main content. Note that this lemma is the result of Lemma 8.

Lemma 9 (Corollary 67 in [4]). Suppose $\gamma, \varepsilon \in (0, \frac{1}{2})$ and $c \in (0, 1)$. Then there exists a polynomial $P \in \mathbb{R}[x]$ of degree $O(\frac{1}{\gamma}\ln\frac{1}{\varepsilon})$ such that for all $x \in [-1, 1]$, $|P(x)| \leqslant 1$, and for all $x \in [\gamma, 1]$, $|P(x) - \frac{\gamma^c}{2}x^{-c}| \leqslant \varepsilon$.

The following is the proof of Theorem 7.

Proof. Reference [70] states that it is possible to obtain an estimation of $S_\alpha(\rho)$ up to precision ε, by an estimation of $\mathrm{tr}(\rho^\alpha)$ within error $\varepsilon' = \frac{|1-\alpha|\mathrm{tr}(\rho^\alpha)}{2}\varepsilon$. Now we turn to demonstrate how to obtain $\mathrm{tr}(\rho^\alpha)$ with error bounded by ε'.

Suppose $\alpha \in (0, 1)$. By Lemma 9 there exists a polynomial $P \in \mathbb{R}[x]$ of degree $O(\frac{1}{\gamma}\ln\frac{1}{\gamma^{1-\alpha}\varepsilon'})$ such that for all $x \in [-1, 1]$, $|P(x)| \leqslant 1$, and for all $x \in [\gamma, 1]$, $|P(x) - \frac{\gamma^{1-\alpha}}{2}x^{\alpha-1}| \leqslant \frac{\gamma^{1-\alpha}}{4}\varepsilon'$. Similar to the proof of Theorem 5, Theorem 4 implies that there exists a QPP circuit $V(\hat{U}_\rho)$ of L layers to estimate $\mathrm{tr}[\rho P(\rho)]$. Up to precision $\frac{\gamma^{1-\alpha}}{4}\varepsilon'$, the experimental estimation E can be obtained by measuring the first qubit of $V(\hat{U}_\rho)$ with input state $|0^{\otimes(m+1)}\rangle\langle 0^{\otimes(m+1)}| \otimes \rho$. We have

$$\left|E - \frac{\gamma^{1-\alpha}}{2}\mathrm{tr}(\rho^\alpha)\right| \leqslant \frac{\gamma^{1-\alpha}}{2}\varepsilon' \tag{F22}$$

and hence

$$|2\gamma^{\alpha-1}E - \mathrm{tr}(\rho^\alpha)| \leqslant \varepsilon'. \tag{F23}$$

By considering Chebyshev's inequality, the total query complexity of U_ρ and $U_\rho^\dagger$ is $O(\frac{1}{\gamma^{3-2\alpha}\varepsilon'^2}\ln\frac{1}{\gamma^{1-\alpha}\varepsilon'})$. Now replace ε' by ε. The query complexity becomes

$$O\left(\frac{\mathrm{tr}(\rho^\alpha)^{-2}}{|1-\alpha|^2\gamma^{3-2\alpha}\varepsilon^2}\ln\left(\frac{\mathrm{tr}(\rho^\alpha)}{\gamma^{1-\alpha}\varepsilon}\right)\right). \tag{F24}$$

Now suppose $\alpha > 1$. Define $f(x) := \frac{1}{2\ln(2e/\gamma)}x^{\alpha-\lfloor\alpha\rfloor}$, which is bounded by $\frac{1}{2\ln(2e/\gamma)}$ for $x \in [0, 1]$. Using the same setup as for the proof of Theorem 5, we would have $B = \frac{1}{2}$. Note the following deductions:

$$\begin{aligned} &\sum_{l=0}^{\infty}\left|\binom{\alpha-\lfloor\alpha\rfloor}{l}\right|(1-\gamma/2)^l \\ &\quad \leqslant \sum_{l=0}^{\infty}\frac{\alpha-\lfloor\alpha\rfloor}{l}(-1+\gamma/2)^l && \text{(F25)} \\ &\quad = 1 - (\alpha-\lfloor\alpha\rfloor)\sum_{l=1}^{\infty}\frac{(-1)^{l-1}}{l}(-1+\gamma/2)^l && \text{(F26)} \\ &\quad = 1 - (\alpha-\lfloor\alpha\rfloor)\ln(\gamma/2) \leqslant \ln(2e/\gamma). && \text{(F27)} \end{aligned}$$

Immediately, Lemma 8 implies that there exists a polynomial $\tilde{P}(x) \in \mathbb{R}[x]$ of degree $O(\frac{1}{\gamma}\ln\frac{\ln(1/\gamma)}{\varepsilon})$ such that for all $x \in [\gamma, 1]$,

$$|\tilde{P}(x) - f(x)| \leqslant \frac{\varepsilon'}{4\ln(2e/\gamma)}. \tag{F28}$$

Redefine $P(x) := x^{\lfloor\alpha\rfloor}\tilde{P}(x)$ of degree $O(\alpha + \frac{1}{\gamma}\ln\frac{\ln(1/\gamma)}{\varepsilon})$. By the same procedure as for the case of $\alpha \in (0, 1)$, we can find a QPP circuit estimating $\mathrm{tr}[\rho P(\rho)]$ so that the experimental estimation E satisfies

$$|2\ln(2e/\gamma)E - \mathrm{tr}(\rho^\alpha)| \leqslant \varepsilon'. \tag{F29}$$

TABLE V. Comparison of algorithms for estimating von Neumann entropy within additive error. Here the $\tilde{O}$ notation omits logarithmic factors, $\gamma > 0$ is the lower bound of eigenvalues, $\kappa > 0$ is the rank of the state $\rho \in \mathbb{C}^{d\times d}$, and ε is the additive error of estimating $S(\rho)$.

Methods for $S(\rho)$ estimation	Total queries to U_ρ and $U_\rho^\dagger$	Queries per use of circuit
QSVT based with QAE (from [11])	$\tilde{O}(\frac{d}{\varepsilon^{1.5}})$	$\tilde{O}(\frac{d}{\varepsilon^{1.5}})$
QSVT based with QAE (assumes rank, from [47])	$\tilde{O}(\frac{\kappa^2}{\varepsilon^2})$	$\tilde{O}(\frac{\kappa^2}{\varepsilon^2})$
QPP based (assumes rank, in Corollary 3)	$\tilde{O}(\frac{\kappa}{\varepsilon^3})$	$\tilde{O}(\frac{\kappa}{\varepsilon})$
QPP based with QAE (assumes rank, in Corollary 3)	$\tilde{O}(\frac{\kappa}{\varepsilon^2})$	$\tilde{O}(\frac{\kappa}{\varepsilon^2})$
QPP based (in Theorem 5)	$\tilde{O}(\frac{1}{\gamma\varepsilon^2})$	$\tilde{O}(\frac{1}{\gamma})$
QPP based with QAE (in Theorem 5)	$\tilde{O}(\frac{1}{\gamma\varepsilon})$	$\tilde{O}(\frac{1}{\gamma\varepsilon})$

As a result, the total query complexity in terms of ε is

$$O\left(\frac{\operatorname{tr}(\rho^\alpha)^{-2}\ln^2(1/\gamma)}{|1-\alpha|^2\gamma\varepsilon^2}\left(\alpha\gamma+\ln\frac{\operatorname{tr}(\rho^\alpha)\ln(1/\gamma)}{\varepsilon}\right)\right). \tag{F30}$$

The statement for using amplitude estimation follows by switching Chebyshev's inequality to the complexity of amplitude estimation. ■

As for the proof of Theorem 8, Theorem 4 implies the query complexity is simply $O(\frac{\alpha}{\varepsilon'^2})$ when α is an integer. Then the statement of Theorem 8 follows by replacing ε' with ε. Note that in this case $|1-\alpha| \in O(\alpha)$. In the following corollary, we present a method to estimate $S_\alpha(\rho)$ when γ is unknown.

Corollary 4. Assume a rank $\kappa \leqslant 2^n$ for an n-qubit state ρ and a purified quantum oracle U_ρ. There exists a QPP circuit that estimates $S_\alpha(\rho)$ within precision ε by measuring a single qubit, at the expense of querying U_ρ and $U_\rho^\dagger$ for a number of

$$\begin{aligned}
&O\left(\frac{2^{12/\alpha}\kappa^{3/\alpha-2}}{\alpha\varepsilon^{3/\alpha}}\left[3+\ln\left(\frac{\operatorname{tr}(\rho^\alpha)\kappa}{|1-\alpha|\varepsilon}\right)\right]\eta^{3/\alpha}\right) \quad \text{if } \alpha\in(0,1),\\
&O\left(\frac{\kappa}{\varepsilon^3}\ln^2\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}\kappa}{|1-\alpha|\varepsilon}\right)\left[\frac{\alpha^2\varepsilon}{2\kappa}+\ln\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}\kappa}{|1-\alpha|\varepsilon}\right)\ln\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}}{|1-\alpha|\varepsilon}\right)\right]\eta^3\right) \quad \text{if } \alpha\in(1,\infty),
\end{aligned} \tag{F31}$$

where $\eta = \frac{\operatorname{tr}(\rho^\alpha)^{-1}}{|1-\alpha|}$. Moreover, using amplitude estimation improves query complexity to

$$\begin{aligned}
&O\left(\frac{2^{8/\alpha}\kappa^{2/\alpha-1}}{\alpha\varepsilon^{2/\alpha}}\left[3+\ln\left(\frac{\operatorname{tr}(\rho^\alpha)\kappa}{|1-\alpha|\varepsilon}\right)\right]\eta^{2/a}\right) \quad \text{if } \alpha\in(0,1),\\
&O\left(\frac{\kappa}{\varepsilon^2}\ln\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}\kappa}{|1-\alpha|\varepsilon}\right)\left[\frac{\alpha^2\varepsilon}{2\kappa}+\ln\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}\kappa}{|1-\alpha|\varepsilon}\right)\ln\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}}{|1-\alpha|\varepsilon}\right)\right]\eta^2\right) \quad \text{if } \alpha\in(1,\infty).
\end{aligned} \tag{F32}$$

Proof. As shown in the proof of Theorem 7, we could find a polynomial $P(x)$ such that

$$|P(x)-f_\alpha(x)| \leqslant \eta\,\forall\, x\in[\gamma,1], \tag{F33}$$

where

$$f_\alpha(x) = \begin{cases}\frac{\gamma^{1-\alpha}}{2}x^{\alpha-1} & \text{if } \alpha\in(0,1)\\ \frac{x^\alpha}{2\ln(2e/\gamma)} & \text{if } \alpha>1.\end{cases} \tag{F34}$$

Denote by E the experiment estimation. Then

$$\begin{aligned}
&|E-\operatorname{tr}[\rho f(\rho)]|\\
&\quad\leqslant |E-\operatorname{tr}[\rho P(\rho)]| + |\operatorname{tr}[\rho P(\rho)]-\operatorname{tr}[\rho f(\rho)]| &\text{(F35)}\\
&\quad\leqslant |E-\operatorname{tr}[\rho P(\rho)]| + \kappa\|xP-xf\|_{[0,\gamma]} + \|P-f\|_{[\gamma,1]} &\text{(F36)}\\
&\quad\leqslant |E-\operatorname{tr}[\rho P(\rho)]| + 2\kappa\gamma + \eta. &\text{(F37)}
\end{aligned}$$

For $\alpha>1$, choose $|E-\operatorname{tr}[\rho P(\rho)]| \leqslant \frac{\varepsilon'}{16\ln(2e/\gamma)}$, $\gamma = \frac{\varepsilon'}{16\kappa\ln(16\kappa/\varepsilon')}$, and $\eta = \frac{\varepsilon'}{16\ln(2e/\gamma)}$. We have

$$|2\ln(2e/\gamma)E-\operatorname{tr}(\rho^\alpha)| \leqslant \frac{\varepsilon'}{4}+\frac{\varepsilon'}{2}+\frac{\varepsilon'}{4} = \varepsilon'. \tag{F38}$$

Then the total complexity is

$$\begin{aligned}
&O\left(\frac{1}{(\varepsilon')^2}\ln^2\left(\frac{\kappa}{\varepsilon'}\right)\right)O\left(\alpha+\frac{\kappa}{\varepsilon'}\ln\left(\frac{\kappa}{\varepsilon'}\right)\ln\left(\frac{1}{\varepsilon'}\right)\right) &\text{(F39)}\\
&\quad= O\left(\frac{\kappa}{(\varepsilon')^3}\ln^2\left(\frac{\kappa}{\varepsilon'}\right)\left[\frac{\alpha\varepsilon'}{\kappa}+\ln\left(\frac{\kappa}{\varepsilon'}\right)\ln\left(\frac{1}{\varepsilon'}\right)\right]\right) &\text{(F40)}\\
&\quad= O\left(\frac{\operatorname{tr}(\rho^\alpha)^{-3}\kappa}{|1-\alpha|^3\varepsilon^3}\ln^2\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}\kappa}{|1-\alpha|\varepsilon}\right)\left[\frac{\alpha|1-\alpha|\operatorname{tr}(\rho^\alpha)\varepsilon}{2\kappa}\right.\right.\\
&\qquad\left.\left.+\ln\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}\kappa}{|1-\alpha|\varepsilon}\right)\ln\left(\frac{\operatorname{tr}(\rho^\alpha)^{-1}}{|1-\alpha|\varepsilon}\right)\right]\right). &\text{(F41)}
\end{aligned}$$

The result follows by the fact that $\alpha|1-\alpha| = O(\alpha^2)$ and $\operatorname{tr}(\rho^\alpha) \leqslant 1$.

For $\alpha\in(0,1)$, choose $|E-\operatorname{tr}[\rho P(\rho)]| \leqslant \frac{\gamma^{1-\alpha}\varepsilon'}{8}$, $\gamma = 8^{-1/\alpha}(\frac{\varepsilon'}{\kappa})^{1/\alpha}$, and $\eta = \frac{\gamma^{1-\alpha}\varepsilon'}{8}$. We have

$$|2\gamma^{\alpha-1}E-\operatorname{tr}(\rho^\alpha)| \leqslant \frac{\varepsilon'}{4}+\frac{\varepsilon'}{2}+\frac{\varepsilon'}{4} = \varepsilon'. \tag{F42}$$

TABLE VI. Comparison of algorithms on estimating quantum α-Rényi entropies for different α, in terms of the query complexity of U_ρ and $U_\rho^\dagger$. Here the $\hat{O}$ notation omits the logarithmic and α factors, $\gamma > 0$ is the lower bound of eigenvalues, $\kappa > 0$ is the rank of a mixed state $\rho \in \mathbb{C}^{d\times d}$, and ε is the additive error of estimating $S_\alpha(\rho)$.

	Total queries to U_ρ and $U_\rho^\dagger$		
Methods for $S_\alpha(\rho)$ estimation	$\alpha \in (0, 1)$	$\alpha \in (1, \infty)$	$\alpha \in \mathbb{N}_+$
QSVT based with DQC1 (from [12])	$\hat{O}\left(\frac{d^2}{\gamma\varepsilon^2}\eta^2\right)$	$\hat{O}\left(\frac{d^2}{\gamma\varepsilon^2}\eta^2\right)$	$\hat{O}\left(\frac{d^2}{\varepsilon^2}\eta^2\right)$
QSVT based with QAE (assumes rank, from [47])	$\tilde{O}\left(\frac{\kappa^{(3-\alpha^2)/2\alpha}}{\varepsilon^{(3/+\alpha)2\alpha}}\right)$	$\tilde{O}\left(\frac{\kappa^{\alpha-1+\alpha/[(\alpha-1)/2]}}{\varepsilon^{1+1/[(\alpha-/1)2]}}\right)$	$\tilde{O}\left(\frac{\kappa^{\alpha-1}}{\varepsilon}\right)$ (only for odd α)
QPP based (assumes rank, in Corollary 4)	$\hat{O}\left(\frac{\kappa^{3/\alpha-2}}{\varepsilon^{3/\alpha}}\eta^{3/\alpha}\right)$	$\hat{O}\left(\frac{\alpha^2\varepsilon+\kappa}{\varepsilon^3}\eta^3\right)$	$\hat{O}\left(\frac{1}{\varepsilon^2}\eta^2\right)$
QPP based with QAE (assumes rank, in Corollary 4)	$\hat{O}\left(\frac{\kappa^{2/\alpha-1}}{\varepsilon^{2/\alpha}}\eta^{2/\alpha}\right)$	$\hat{O}\left(\frac{\alpha^2\varepsilon+\kappa}{\varepsilon^2}\eta^2\right)$	$\hat{O}\left(\frac{1}{\varepsilon}\eta\right)$
QPP based (in Theorems 7 and 8)	$\hat{O}\left(\frac{1}{\gamma^{3-2\alpha}\varepsilon^2}\eta^2\right)$	$\hat{O}\left(\frac{\alpha\gamma+1}{\gamma\varepsilon^2}\eta^2\right)$	$\hat{O}\left(\frac{1}{\varepsilon^2}\eta^2\right)$
QPP based with QAE (in Theorems 7 and 8)	$\hat{O}\left(\frac{1}{\gamma^{2-\alpha}\varepsilon}\eta\right)$	$\hat{O}\left(\frac{\alpha\gamma+1}{\gamma\varepsilon}\eta\right)$	$\hat{O}\left(\frac{1}{\varepsilon}\eta\right)$

Then the total complexity is

$$O\left(\frac{\gamma^{2(\alpha-1)}}{(\varepsilon')^2}\right)O\left(\frac{1}{\gamma}\ln\left(\frac{\gamma^{\alpha-1}}{\varepsilon'}\right)\right) \tag{F43}$$

$$= O\left(\frac{2^{6/\alpha}\kappa^{2/\alpha-2}}{(\varepsilon')^{2/\alpha}}\right)O\left(\frac{2^{3/\alpha}\kappa^{1/\alpha}}{(\varepsilon')^{1/\alpha}}\ln\left(\frac{8^{1/\alpha-1}\kappa^{1/\alpha-1}}{(\varepsilon')^{1/\alpha}}\right)\right)$$

$$= O\left(\frac{2^{9/\alpha}\kappa^{3/\alpha-2}}{\alpha(\varepsilon')^{3/\alpha}}\left[3+\ln\left(\frac{\kappa}{\varepsilon'}\right)\right]\right) \tag{F44}$$

$$= O\left(\frac{2^{12/\alpha}\mathrm{tr}(\rho^\alpha)^{-3/\alpha}\kappa^{3/\alpha-2}}{\alpha|1-\alpha|^{3/\alpha}\varepsilon^{3/\alpha}}\left[3+\ln\left(\frac{\mathrm{tr}(\rho^\alpha)\kappa}{|1-\alpha|\varepsilon}\right)\right]\right), \tag{F45}$$

as required. The statement for using amplitude estimation follows by switching Chebyshev's inequality to the complexity of amplitude estimation. ■

4. Comparison of entropy estimation algorithms

We present further comparison of different entropy estimation algorithms in this section. In Table V we add the von Neumann entropy estimation algorithm proposed by Wang *et al.* [47]. When assuming rank κ, the QPP-based algorithm improves the result in [47] by a factor of κ. In Table VI, we add Rényi entropy estimation algorithms proposed by Wang *et al.* [47].

[1] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM J. Comput. **26**, 1484 (1997).

[2] L. K. Grover, *Proceedings of the 28th Annual ACM Symposium on Theory of Computing* (Association for Computing Machinery, New York, 1996), pp. 212–219.

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

[4] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* (ACM, New York, 2019), pp. 193–204.

[5] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, A grand unification of quantum algorithms, PRX Quantum **2**, 040203 (2021).

[6] P. Rall, Faster coherent quantum algorithms for phase, energy, and amplitude estimation, Quantum **5**, 566 (2021).

[7] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum **3**, 163 (2019).

[8] S. Lloyd, B. T. Kiani, D. R. M. Arvidsson-Shukur, S. Bosch, G. De Palma, W. M. Kaminsky, Z.-W. Liu, and M. Marvian, Hamiltonian singular value transformation and inverse block encoding, arXiv:2104.01410.

[9] J. M. Martyn, Y. Liu, Z. E. Chin, and I. L. Chuang, Efficient fully-coherent quantum signal processing algorithms for real-time dynamics simulation, J. Chem. Phys. **158**, 024106 (2023).

[10] A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, Toward the first quantum simulation with quantum speedup, Proc. Natl. Acad. Sci. USA **115**, 9456 (2018).

[11] A. Gilyén and T. Li, in *11th Innovations in Theoretical Computer Science Conference, Seattle, 2020*, edited by T. Vidick (Schloss Dagstuhl–Leibniz-Zentrum für Informatik GmbH, Saarbrücken, 2020), pp. 25:1–25:19.

[12] S. Subramanian and M.-H. Hsieh, Quantum algorithm for estimating α-Renyi entropies of quantum states, Phys. Rev. A **104**, 022428 (2021).

[13] T. Gur, M.-H. Hsieh, and S. Subramanian, Sublinear quantum algorithms for estimating von Neumann entropy, arXiv:2111.11139.

[14] A. Gilyén and A. Poremba, Improved quantum algorithms for fidelity estimation, arXiv:2203.15993.

[15] L. Lin and Y. Tong, Near-optimal ground state preparation, Quantum **4**, 372 (2020).

[16] L. Lin and Y. Tong, Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers, PRX Quantum **3**, 010318 (2022).

[17] Y. Dong, L. Lin, and Y. Tong, Ground state preparation and energy estimation on early fault-tolerant quantum computers via quantum eigenvalue transformation of unitary matrices, PRX Quantum **3**, 040305 (2022).

[18] G. H. Low, T. J. Yoder, and I. L. Chuang, Methodology of resonant equiangular composite quantum gates, Phys. Rev. X **6**, 041067 (2016).

[19] G. H. Low and I. L. Chuang, Optimal Hamiltonian simulation by quantum signal processing, Phys. Rev. Lett. **118**, 010501 (2017).

[20] A. M. Childs and N. Wiebe, Hamiltonian simulation using linear combinations of unitary operations, Quantum Inf. Comput. **12**, 901 (2012).

[21] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, Contemp. Math. **305**, 53 (2002).

[22] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, *Proceedings of the 46th Annual ACM Symposium on Theory of Computing* (Association for Computing Machinery, New York, 2014), pp. 283–292.

[23] J. Haah, Product decomposition of periodic functions in quantum signal processing, Quantum **3**, 190 (2019).

[24] R. Chao, D. Ding, A. Gilyen, C. Huang, and M. Szegedy, Finding angles for quantum signal processing with machine precision, arXiv:2003.02831.

[25] Z. Yu, H. Yao, M. Li, and X. Wang, in *Advances in Neural Information Processing Systems 35*, edited by S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Curran, Red Hook, 2022).

[26] A. Y. Kitaev, Quantum measurements and the Abelian stabilizer problem, arXiv:quant-ph/9511026.

[27] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Quantum algorithms revisited, Proc. R. Soc. London A **454**, 339 (1998).

[28] A. Barenco, A. Berthiaume, D. Deutsch, A. Ekert, R. Jozsa, and C. Macchiavello, Stabilization of quantum computations by symmetrization, SIAM J. Comput. **26**, 1541 (1997).

[29] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf, Quantum fingerprinting, Phys. Rev. Lett. **87**, 167902 (2001).

[30] D. Aharonov, V. Jones, and Z. Landau, A polynomial quantum algorithm for approximating the Jones polynomial, Algorithmica **55**, 395 (2009).

[31] E. Knill and R. Laflamme, Power of one bit of quantum information, Phys. Rev. Lett. **81**, 5672 (1998).

[32] D. Petz, Quasi-entropies for finite quantum systems, Rep. Math. Phys. **23**, 57 (1986).

[33] T. d. L. Silva, L. Borges, and L. Aolita, Fourier-based quantum signal processing, arXiv:2206.02826.

[34] Y. Dong, X. Meng, K. B. Whaley, and L. Lin, Efficient phase-factor evaluation in quantum signal processing, Phys. Rev. A **103**, 042419 (2021).

[35] D. Grinko, J. Gacon, C. Zoufal, and S. Woerner, Iterative quantum amplitude estimation, npj Quantum Inf. **7**, 52 (2021).

[36] A. Kitaev, A. Shen, and M. Vyalyi, *Classical and Quantum Computation*, Graduate Studies in Mathematics Vol. 47 (American Mathematical Society, Providence, 2002), p. 272.

[37] D. Coppersmith, An approximate Fourier transform useful in quantum factoring, arXiv:quant-ph/0201067.

[38] R. B. Griffiths and C.-S. Niu, Semiclassical Fourier transform for quantum computation, Phys. Rev. Lett. **76**, 3228 (1996).

[39] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2010).

[40] A. Belovs, in *27th Annual European Symposium on Algorithms, Munich, 2019*, edited by M. A. Bender, O. Svensson and G. Herman (Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Saarbrücken, 2019), pp. 16:1–16:11.

[41] J. von Neumann, *Mathematische Grundlagen der Quantenmechanik*, 2nd ed., Die Grundlehren der mathematischen Wissenschaften in Einzeldarstellungen No. 38 (Springer, Berlin, 1996).

[42] A. Rényi, *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability* (University of California Press, Berkeley, 1961), Vol. 1, pp. 547–561.

[43] S. Johri, D. S. Steiger, and M. Troyer, Entanglement spectroscopy on a quantum computer, Phys. Rev. B **96**, 195136 (2017).

[44] Y. Subasi, L. Cincio, and P. J. Coles, Entanglement spectroscopy with a depth-two quantum circuit, J. Phys. A: Math. Theor. **52**, 044001 (2019).

[45] J. Yirka and Y. Subasi, Qubit-efficient entanglement spectroscopy using qubit resets, Quantum **5**, 535 (2021).

[46] I. Kerenidis and A. Prakash, in *8th Innovations in Theoretical Computer Science Conference, Berkeley, 2017*, edited by C. H. Papadimitriou (Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Saarbrücken, 2017), pp. 49:1–49:21.

[47] Q. Wang, J. Guan, J. Liu, Z. Zhang, and M. Ying, New quantum algorithms for computing quantum entropies and distances, arXiv:2203.13522.

[48] S. Chakraborty, A. Gilyén, and S. Jeffery, Proceedings of the 46th International Colloquium on Automata, Languages, and Programming, Patras, 2019 (Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Saarbrücken, 2019), Vol. 132, pp. 33:1–33:14.

[49] A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, and J. I. Latorre, Data re-uploading for a universal quantum classifier, Quantum **4**, 226 (2020).

[50] M. Cerezo, G. Verdon, H.-Y. Huang, L. Cincio, and P. J. Coles, Challenges and opportunities in quantum machine learning, Nat. Comput. Sci. **2**, 567 (2022).

[51] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Quantum machine learning, Nature (London) **549**, 195 (2017).

[52] G. Li, R. Ye, X. Zhao, and X. Wang, in *Advances in Neural Information Processing Systems 35* (Ref. [25]).

[53] Y. Du, Y. Yang, D. Tao, and M.-H. Hsieh, Problem-dependent power of quantum neural networks on multiclass classification, Phys. Rev. Lett. **131**, 140601 (2023).

[54] X. Wang, Z. Song, and Y. Wang, Variational quantum singular value decomposition, Quantum **5**, 483 (2021).

[55] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, Nat. Rev. Phys. **3**, 625 (2021).

[56] J. Romero, J. P. Olson, and A. Aspuru-Guzik, Quantum autoencoders for efficient compression of quantum data, Quantum Sci. Technol. **2**, 045001 (2017).

[57] Z. Yu, X. Zhao, B. Zhao, and X. Wang, Optimal quantum dataset for learning a unitary transformation, Phys. Rev. Appl. **19**, 034017 (2023).

[58] M. Larocca, F. Sauvage, F. M. Sbahi, G. Verdon, P. J. Coles, and M. Cerezo, Group-invariant quantum machine learning, PRX Quantum **3**, 030341 (2022).

[59] J. Liu, K. Najafi, K. Sharma, F. Tacchino, L. Jiang, and A. Mezzacapo, Analytic theory for the dynamics of wide quantum neural networks, Phys. Rev. Lett. **130**, 150601 (2023).

[60] R. Rines and I. Chuang, High performance quantum modular multipliers, arXiv:1801.01081.

[61] S.-M. Cho, A. Kim, D. Choi, B.-S. Choi, and S.-H. Seo, Quantum modular multiplication, IEEE Access **8**, 213244 (2020).

[62] Y. Suzuki, S. Uno, R. Raymond, T. Tanaka, T. Onodera, and N. Yamamoto, Amplitude estimation without phase estimation, Quantum Inf. Process. **19**, 75 (2020).

[63] S. Aaronson and P. Rall, *Symposium on Simplicity in Algorithms* (SIAM, Philadelphia, 2020), pp. 24–32.

[64] P. Rall and B. Fuller, Amplitude estimation from quantum signal processing, Quantum **7**, 937 (2023).

[65] D. Camps, L. Lin, R. Van Beeumen, and C. Yang, Explicit quantum circuits for block encodings of certain sparse matrices, arXiv:2203.10236.

[66] M. Abramowitz, I. A. Stegun, and R. H. Romer, Handbook of mathematical functions with formulas, graphs, and mathematical tables, Am. J. Phys. **56**, 958 (1988).

[67] D. W. Berry, A. M. Childs, and R. Kothari, *Proceedings of the IEEE 56th Annual Symposium on Foundations of Computer Science, Berkeley, 2015* (IEEE, Piscataway, 2015), pp. 792–809.

[68] D. S. Abrams and S. Lloyd, Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors, Phys. Rev. Lett. **83**, 5162 (1999).

[69] D. Poulin, A. Kitaev, D. S. Steiger, M. B. Hastings, and M. Troyer, Quantum algorithm for spectral measurement with a lower gate count, Phys. Rev. Lett. **121**, 010501 (2018).

[70] Y. Wang, B. Zhao, and X. Wang, Quantum algorithms for estimating quantum entropies, Phys. Rev. Appl. **19**, 044041 (2023).