

Gravitational wave matched filtering by quantum Monte Carlo integration and quantum amplitude amplification

Koichi Miyamoto ^{1,*}, Gonzalo Morras ^{2,†}, Takahiro S. Yamamoto ^{3,‡}, Sachiko Kuroyanagi ^{2,3,§} and Savvas Nesseris ^{2,||}

¹Center for Quantum Information and Quantum Biology, Osaka University, Toyonaka 560-0043, Japan

²Instituto de Física Teórica UAM-CSIC, Universidad Autónoma de Madrid, Cantoblanco, 28049 Madrid, Spain

³Department of Physics and Astrophysics, Nagoya University, Nagoya 464-8602, Japan



(Received 19 May 2022; accepted 10 August 2022; published 26 August 2022)

The speedup of heavy numerical tasks by quantum computing is now actively investigated in various fields including data analysis in physics and astronomy. In this paper, we propose a quantum algorithm for matched filtering in gravitational wave (GW) data analysis based on the previous work by Gao *et al.* [*Phys. Rev. Research* **4**, 023006 (2022)]. Our approach uses the quantum algorithm for Monte Carlo integration for the signal-to-noise ratio (SNR) calculation instead of the fast Fourier transform used in Gao *et al.* and searches signal templates with high SNR by quantum amplitude amplification. In this way, we achieve an exponential reduction of the qubit number compared with Gao *et al.*'s algorithm, keeping a quadratic speedup over classical GW matched filtering with respect to the template number.

DOI: [10.1103/PhysRevResearch.4.033150](https://doi.org/10.1103/PhysRevResearch.4.033150)

I. INTRODUCTION

Quantum computing [1] is a developing technology and is expected to speed up some classes of computation that are intractable in classical computing. The recent rapid advance in quantum computer development has been stimulating research on applications of quantum algorithms to concrete problems in various fields (see Ref. [2] for a recent review).

In this paper, we study an application of some quantum algorithms to a problem in gravitational wave (GW) experiments, which measures the space-time distortion caused by GWs with a laser interferometer. A recent paper [3] has proposed the use of Grover's search algorithm for matched filtering, which is a commonly used technique to search a signal buried in noisy data and is widely used in GW data analysis [4–7]. Given a target signal waveform, called a template, we take an inner product between a template and data to cancel out the noise contribution and extract a signal. In fact, the first GW was detected by the LIGO detectors in 2015 [8], and, after that, the worldwide GW detector network has observed tens of GW events using matched filtering [9–11].

A challenging point in GW matched filtering is the large number of templates. The functional form of a GW signal is

predicted by general relativity depending on the GW source such as compact binary coalescence (CBC) [4–7], but it has some parameters such as masses of the compact objects, spin parameters, and luminosity distance. In order to get a high signal-to-noise ratio (SNR), we must perform matched filtering using a template with appropriate parameters. Therefore, usually, we set sufficiently many points in the parameter space and run an exhaustive search. That is, we repeat matched filtering using templates one by one to find those that yield SNR larger than some threshold. This is an extremely time-consuming task and expected to be sped up by quantum computing.

Fortunately, there exists a quantum algorithm for searching, called Grover's algorithm [12]. Given N data, x_1, \dots, x_N , represented as bit strings and a condition F as a function that maps a bit string to 0 or 1, Grover's algorithm can find “marked data” x such that $F(x) = 1$ making $O(\sqrt{N/n})$ calls to F , where n is the number of marked data. Therefore, it is often said that Grover's algorithm provides a quadratic speedup over the classical exhaustive search, which has $O(N/n)$ query complexity. As an application of this, Ref. [3] presented a quantum algorithm for GW matched filtering. With the SNR calculation implemented as a quantum circuit, the aforementioned algorithm can find a template with SNR higher than a threshold ρ_{th} with $\tilde{O}(M/\sqrt{r(\rho_{\text{th}})})$ complexity,¹ where $r(\rho_{\text{th}})$ is the fraction of templates that yield SNR $\rho \geq \rho_{\text{th}}$ and M is the number of points in the time-series data of the detector output, or, equivalently, the number of frequency bins of Fourier transformed data. This is in fact a quadratic speedup over the classical method, which has $O(M/r(\rho_{\text{th}}))$ complexity, with respect to the template number.

*miyamoto.kouichi.qiqb@osaka-u.ac.jp

†gonzalo.morras@estudiante.uam.es

‡yamamoto.takahiro@f.mbox.nagoya-u.ac.jp

§sachiko.kuroyanagi@csic.es

||savvas.nesseris@csic.es

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

¹In the big-O notation, we use a symbol $\tilde{O}(\cdot)$, which hides logarithmic factors in $O(\cdot)$

However, this quantum algorithm has the following subtlety. It uses fast Fourier transform (FFT) [13] for the SNR calculation, which is also used in the usual classical way. FFT simultaneously calculates SNR for M possible values of a parameter called time of coalescence, with other parameter fixed, in $O(M \log M)$ time, whereas naively such a computation takes $O(M^2)$ time without FFT. However, in order to store the intermediate and final calculation results, FFT requires $O(M)$ qubits, which is a somewhat large number since M is typically of order $4096 \text{ Hz} \times 256 \text{ s} \sim O(10^6)$, where 4096 Hz is the sampling frequency and 256 s is the typical duration of the data segment [7]. This might cause an issue on feasibility, since fault-tolerant quantum computers will have a limitation on the number of qubits available even in the future. It is expected that creating one logical qubit requires thousands or tens of thousands physical qubits for error correction [14], and therefore realizing a quantum computer with millions of qubits is very challenging.

In light of this, we propose an alternative quantum algorithm for GW matched filtering, in which the SNR calculation with FFT is replaced with the quantum algorithm for Monte Carlo integration (QMCI) [15]. QMCI is a method to estimate an integral given in the finite sum approximation. Thus, it can be applied to the calculation of SNR, which includes frequency band integration and is in practice calculated as a sum of contributions from many Fourier modes. In this approach, the required qubit number scales on M as $O(\text{poly}(\log M))$, which means an exponential reduction from the FFT approach. Note that this is not just a straightforward application of another quantum algorithm to a part of an existing method, since the use of QMCI causes the following issue. Unlike FFT, which calculates SNR deterministically, QMCI inevitably accompanies errors, and thus comparing the SNR calculated by QMCI with a single SNR threshold ρ_{th} leads to a false alarm that the detector output yields SNR larger than ρ_{th} for some templates despite there being no such event. As a solution to this, we propose to set two thresholds ρ_{hard} and ρ_{soft} that have the following meanings: We should never miss events with SNR $\rho \geq \rho_{\text{hard}}$, and we do not want to be falsely alarmed by events with $\rho < \rho_{\text{soft}}$. Then, with QMCI accuracy set according to the difference between ρ_{hard} and ρ_{soft} , the proposed algorithm says “there is a signal” for events with SNR $\rho \geq \rho_{\text{hard}}$ with high probability, “there is no signal” for events with SNR $\rho < \rho_{\text{soft}}$ with certainty, and either of these messages for events with SNR $\rho \in [\rho_{\text{soft}}, \rho_{\text{hard}})$. The query complexity in this algorithm is of order $O(M/\sqrt{r(\rho_{\text{hard}})})$, which still indicates a quadratic speedup.

The remaining part of this paper is organized as follows. Section II introduces the preliminary knowledge. We outline GW matched filtering and some building-block quantum algorithms such as Grover’s algorithm, quantum amplitude amplification (QAA), quantum amplitude estimation (QAE), and QMCI. Section III is the main part. Defining GW matched filtering as a mathematical problem, we explain the existing algorithm in [3], and present our modified algorithm in detail, along with estimation of the query complexity and the qubit number and a plausible setting on thresholds ρ_{hard} and ρ_{soft} . Section IV summarizes this paper. Some proofs are presented in appendices.

II. PRELIMINARY

A. Notation

Here, we summarize some notations used in this paper. \mathbb{R}_+ denotes the set of all positive real numbers. For $n \in \mathbb{N}$, we define $[n] := \{1, \dots, n\}$ and $[n]_0 := \{0, 1, \dots, n-1\}$. For any $x \in \mathbb{R}$, if $|x - y| \leq \epsilon$ holds for some $y \in \mathbb{R}$ and $\epsilon \in \mathbb{R}_+$, we say that x is ϵ close to y and that x is an ϵ approximation of y . For any equation or inequality C , $\mathbb{1}_C$ takes 1 if C is satisfied, and 0 otherwise. For $c \in \mathbb{C}$, c^* denotes its complex conjugate. For $\mathcal{X} = \{x_1, \dots, x_n\}$ and $\mathcal{Y} = \{y_1, \dots, y_n\}$, finite sets of real numbers with same size n , we define the sample mean $\text{Mean}(\mathcal{X}) := \frac{1}{n} \sum_{i=1}^n x_i$, the sample variance $\text{Var}(\mathcal{X}) := \frac{1}{n} \sum_{i=1}^n (x_i - \text{Mean}(\mathcal{X}))^2$ and the sample covariance $\text{Cov}(\mathcal{X}, \mathcal{Y}) := \frac{1}{n} \sum_{i=1}^n (x_i - \text{Mean}(\mathcal{X}))(y_i - \text{Mean}(\mathcal{Y}))$. For $n \in \mathbb{N}$, \mathbb{I}_n denotes the $n \times n$ identity matrix. For $z \in \mathbb{C}$, $\Re z$ and $\Im z$ are the real and imaginary parts of z .

B. Gravitational wave matched filtering

Here, we outline matched filtering in GW search experiments [4–7]. Suppose that we are given the detector output $s(t)$ as a function of time t , which is a sum of the signal $h(t)$ and the noise $n(t)$,

$$s(t) = h(t) + n(t). \quad (1)$$

We assume that the noise is Gaussian, which means that, for each $f \in \mathbb{R}_+$, $\Re \tilde{n}(f)$ and $\Im \tilde{n}(f)$ are normal random variables and

$$\mathbb{E}_n[\tilde{n}(f)\tilde{n}^*(f')] = \frac{1}{2}S_n(|f|)\delta(f - f') \quad (2)$$

holds with the single-sided power spectrum density (PSD) S_n . Here, for any function $q(t)$ in time domain, $\tilde{q}(f) := \int_{-\infty}^{\infty} dt e^{2\pi i f t} q(t)$ is its Fourier transform, $\mathbb{E}_n[\cdot]$ denotes an expectation with respect to randomness of the noise and $\delta(\cdot)$ is the Dirac delta function. We define the inner product of two functions $q(t)$ and $q'(t)$ in time domain as

$$(q|q') := 4\Re \left(\int_0^{\infty} df \frac{\tilde{q}^*(f)\tilde{q}'(f)}{S_n(f)} \right). \quad (3)$$

The matched filtering search is performed by taking an inner product of $s(t)$ and an appropriate filter function $Q(t)$ that yields a large inner product with the targeted signal. The function $Q(t)$ is often called a template and normalized as $(Q|Q) = 1$. The template bank, the collection of templates, is prepared based on theoretically predicted waveform of signals. The SNR for a detector output s and template Q is then defined as

$$\rho = \frac{(Q|s)}{\sqrt{\mathbb{E}_n[|(Q|n)|^2]}} = 4\Re \left(\int_0^{\infty} df \frac{\tilde{Q}^*(f)\tilde{s}(f)}{S_n(f)} \right). \quad (4)$$

By setting a SNR threshold ρ_{th} , exhaustive search is performed to find a template that gives a SNR larger than ρ_{th} to claim a detection. In reality, nonstationary detector noise known as glitches can generate large values of ρ and cause false detections. To mitigate the effect of glitches, what is done in practice is to modify the variable used to rank the events. Instead of the SNR, real GW searches use more complicated ranking statistics, which include signal consistency tests such as χ^2 [16]. In this paper, we will only consider

the SNR as the ranking statistic for simplicity, but the ideas of our quantum algorithm could be applied to searches with more complicated ranking statistics. Hereafter, we write each Fourier transformed template as $\tilde{Q}_m(f)e^{-2\pi if t_0}$, where t_0 is the time of coalescence and the dependency on other parameters (intrinsic parameters) is put into \tilde{Q}_m . Here, we assume that there are N_{temp} candidates of the intrinsic parameter set in the template bank and label the functions \tilde{Q}_m by $m \in [N_{\text{temp}}]_0$.

In reality, we have a detector output as a sequence of discrete points in time [7]. Suppose that a sequence of $s(t)$ is given by $\{s(\tau_l)\}_{l=0, \dots, M-1}$ at M time points $\tau_0 = 0, \tau_1 = \Delta t, \dots, \tau_{M-1} = (M-1)\Delta t$ with interval Δt . In such a situation, Fourier transforms are given in the discrete form: for each $k \in \{0, 1, \dots, M-1\}$. Thus, we redefine \tilde{s} as

$$\tilde{s}(f_k) := \Delta t \sum_{l=0}^{M-1} s(\tau_l) e^{2\pi i k l / M}, \quad (5)$$

where $f_k = k/T$ and $T = M\Delta t$, and similar quantities $\tilde{h}(f_k)$ and $\tilde{n}(f_k)$ for $h(t)$ and $n(t)$. $\Re \tilde{n}(f_k)$ and $\Im \tilde{n}(f_k)$ are still normal random variables but Eq. (2) is now converted into [17]

$$\mathbb{E}_n[\tilde{n}(f_k) \tilde{n}^*(f_l)] = \frac{1}{2} S_n(f_k) T \delta_{k,l}, \quad (6)$$

where $\delta_{k,l}$ is the Kronecker delta. Then, the SNR now becomes

$$\rho_{m,j} = \frac{4}{M\Delta t} \Re \left(\sum_{k=1}^{M/2-1} \frac{\tilde{Q}_m^*(f_k) \tilde{s}(f_k)}{S_n(f_k)} e^{2\pi i j k / M} \right), \quad (7)$$

for the m th intrinsic parameter set and the time of coalescence given as $t_0 = j\Delta t$ with $j \in [M]_0$.² Here and hereafter, we assume that M is even. In the usual way of classical computing, although it seemingly takes $O(M^2)$ computational time to compute Eq. (7) for all $j \in [M]_0$ with m fixed, we can do this in $O(M \log M)$ time using FFT [13]. This means that, for a fixed intrinsic parameter set, we can quickly search the optimal t_0 in $\{j\Delta t\}_{j=0, \dots, M-1}$ and obtain the optimal SNR $\rho_m := \max_{j \in [M]_0} \rho_{m,j}$, which is why t_0 is dealt with separately from other template parameters in the conventional GW data analysis. On the other hand, for the other intrinsic parameters, we calculate ρ_m for each $m \in [N_{\text{temp}}]_0$ one by one until we get $\rho_m \geq \rho_{\text{th}}$, which results in the number of floating-point operations of order

$$O\left(\frac{M \log M}{r(\rho_{\text{th}})}\right) \quad (8)$$

for $r(\rho_{\text{th}}) > 0$. Here

$$r(\rho) := \frac{|\{m \in [N_{\text{temp}}]_0 \mid \rho_m \geq \rho\}|}{N_{\text{temp}}} \quad (9)$$

is the fraction of intrinsic parameter sets in the template banks that yields SNRs larger than ρ , with time of coalescence optimized. When $r(\rho_{\text{th}}) = 0$, we have to go through all the

template and thus the number of floating-point operations is

$$O(N_{\text{temp}} M \log M). \quad (10)$$

Unlike the above, as we will see later, we do not use FFT in the proposed quantum method. For later convenience, we rewrite Eq. (7) as

$$\begin{aligned} \rho_{m,j} &= \frac{2}{M} \sum_{k=1}^{M/2-1} \tilde{\rho}_{m,j,k}, \\ \tilde{\rho}_{m,j,k} &:= \Re \left(\frac{2\tilde{Q}_m^*(f_k) \tilde{s}(f_k)}{S_n(f_k) \Delta t} \right) \cos \left(\frac{2\pi j k}{M} \right) \\ &\quad - \Im \left(\frac{2\tilde{Q}_m^*(f_k) \tilde{s}(f_k)}{S_n(f_k) \Delta t} \right) \sin \left(\frac{2\pi j k}{M} \right), \end{aligned} \quad (11)$$

and set $\tilde{\rho}_{m,j,0} = 0$.

C. Representation of real numbers on qubits and some basic quantum circuits

In numerical calculations in this paper, we use bit strings on qubits as fixed-point binary representations of real numbers and, for $x \in \mathbb{R}$, we denote by $|x\rangle$ the computational basis state on a quantum register, which corresponds to x . Unless otherwise stated, we assume that each register has N_{dig} qubits, where N_{dig} is a sufficiently large positive integer set independently from parameters in problems under consideration, and neglect errors from finite-precision representation.

For computing with real numbers, we use the quantum circuits for four basic arithmetic operations: addition $O_{\text{add}}|x\rangle|y\rangle|0\rangle = |x\rangle|y\rangle|x+y\rangle$, subtraction $O_{\text{sub}}|x\rangle|y\rangle|0\rangle = |x\rangle|y\rangle|x-y\rangle$, multiplication $O_{\text{mul}}|x\rangle|y\rangle|0\rangle = |x\rangle|y\rangle|xy\rangle$, and division $O_{\text{div}}|x\rangle|y\rangle|0\rangle = |x\rangle|y\rangle|x/y\rangle$, where x, y are any real numbers ($y \neq 0$ for division) and some ancillary registers may be undisplayed. In fact, concrete circuit implementations for these operation have been presented [18–33]. These circuits enable the calculation of rational functions. We also use quantum circuits for calculation of elementary functions f such as exponential, trigonometric functions, and so on: $O_f|x\rangle|0\rangle = |x\rangle|f(x)\rangle$ for any $x \in \mathbb{R}$. Such circuits can be implemented through, for example, piecewise polynomial approximation [34].

In addition to these circuits for numerical calculation, we now list some oracles used in the proposed quantum algorithm. A comparer O_{comp} acts as $O_{\text{comp}}|x\rangle|y\rangle|0\rangle = |x\rangle|y\rangle(\mathbb{1}_{x \geq y}|1\rangle + \mathbb{1}_{x < y}|0\rangle)$ for any $x, y \in \mathbb{R}$. This is actually equivalent to subtraction $x - y$, since, if we adopt 2's complement method to represent negative numbers, the most significant digit represents the sign of a number [35]. A Y rotation with controlled angle gate O_{CY} acts as $O_{\text{CY}}|\theta\rangle = |\theta\rangle \otimes R_Y(\theta)|\psi\rangle$, where

$$R_Y(\theta) := \begin{pmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix},$$

for any $\theta \in \mathbb{R}$ and any single-qubit state $|\psi\rangle$. This is implemented as a sequence of fixed-angle controlled Y-rotation gates $|0\rangle\langle 0| \otimes \mathbb{I}_2 + |1\rangle\langle 1| \otimes R_Y(\theta)$ [36], where $\theta \in \mathbb{R}$ is prefixed. We also use the oracle O_N^{EqPr} , where $N \in \mathbb{N}$, to generate equiprobable superposition of states from $|0\rangle$ to $|N-1\rangle$:

²The sum in Eq. (7) runs over $k \in [\frac{M}{2} - 1]$ rather than $k \in [M]_0$ because we use the one-sided power spectral density as defined in Eq. (2). See [7] for the detail.

$O_N^{\text{EqPr}}|0\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle$. If $N = 2^n$ with some $n \in \mathbb{N}$, we can generate such a state by operating a Hadamard gate on each qubit in a n -qubit register. Also for N that is not a power of 2, we can implement O_N^{EqPr} by the method in [37] to generate a state in which a given probability density $p(x)$ is amplitude-encoded, with $p(x)$ set to the uniform density on $[0, (N - 1)/2^n]$ with $n = \lceil \log_2 N \rceil$.

The last oracle we mention here is O_N^{med} that, for any N real numbers x_1, \dots, x_N , outputs the median $\text{med}(x_1, \dots, x_N)$: $O_N^{\text{med}}|x_1\rangle \cdots |x_N\rangle|0\rangle = |x_1\rangle \cdots |x_N\rangle|\text{med}(x_1, \dots, x_N)\rangle$. This operation is implemented as follows. First, we transform $|x_1\rangle \cdots |x_N\rangle$ to $|x_1^{\text{sort}}\rangle \cdots |x_N^{\text{sort}}\rangle$, where $x_1^{\text{sort}}, \dots, x_N^{\text{sort}}$ is a sequence made by ascending sort of x_1, \dots, x_N and some ancillary qubits are not displayed.³ Then, we let the number on the midmost register be $\text{med}(x_1, \dots, x_N)$. Note that exchange based sort algorithms such as bubble sort [38] can be implemented since the operation

$$|x\rangle|y\rangle|0\rangle \rightarrow |x\rangle|y\rangle|\mathbb{1}_{x \geq y}\rangle \rightarrow \begin{cases} |y\rangle|x\rangle|1\rangle; & \text{if } x \geq y \\ |x\rangle|y\rangle|0\rangle; & \text{otherwise} \end{cases} \quad (12)$$

is possible for any $x, y \in \mathbb{R}$, where we use O_{comp} and controlled SWAP gates at the first and second arrows, respectively.

Hereafter, we collectively call the above oracles arithmetic oracles.

D. Grover’s algorithm and quantum amplitude amplification

Grover’s algorithm [12] is a quantum algorithm for searching a “marked entry” x , which satisfies some condition given as a binary-valued function, from an unstructured database. Formally, we have the following theorem.

Theorem 1. Let $n \in \mathbb{N}$ and $F : \{0, 1\}^n \rightarrow \{0, 1\}$ is a function such that $F(x_{\text{tar}}) = 1$ for one element $x_{\text{tar}} \in \{0, 1\}^n$ and $F(x) = 0$ for any $x \in \{0, 1\}^n \setminus \{x_{\text{tar}}\}$. Suppose that we are given an access to an oracle O_F on a system consisting of a n -qubit register and a qubit such that $O_F|x\rangle|0\rangle = O_F|x\rangle|F(x)\rangle$ for any $x \in \{0, 1\}^n$. Then, for any $\delta \in (0, 1)$, there exists a quantum algorithm that, with probability at least $1 - \delta$, outputs x_{tar} making $O(\sqrt{N} \log \delta^{-1})$ calls to O_F , where $N := 2^n$.

This is often called a quadratic speedup over classical search methods that takes $O(N)$ time for the same problem.

Besides, there exists a quantum algorithm called QAA [39,40], which can be seen as an extension of Grover’s algorithm. It is an algorithm to amplify the amplitude of the “marked state” in a given superposition and obtain the state. Formally, the following theorem holds.

Theorem 2. Suppose that we are given an access to an oracle A that acts on a system consisting of a n -qubit register and a single-qubit register as

$$A|0\rangle|0\rangle = \sqrt{a}|\phi_1\rangle|1\rangle + \sqrt{1-a}|\phi_0\rangle|0\rangle =: |\Phi\rangle, \quad (13)$$

where $|\phi_0\rangle$ and $|\phi_1\rangle$ are some quantum states on the register and $a \in [0, 1)$. Then, for any $\gamma, \delta \in (0, 1)$, there exists a

quantum algorithm $\text{QAA}(A, \gamma, \delta)$ that uses $O(n)$ qubits and behaves as follows:

- (i) The output of the algorithm is either of
 - (A) the message “success” and the quantum state $|\phi_1\rangle$
 - (B) the message “failure”
- (ii) If $a \geq \gamma$, the algorithm outputs (A) with probability at least $1 - \delta$, making $O(\frac{\log \delta^{-1}}{\sqrt{a}})$ queries to A .
- (iii) If $a < \gamma$, the algorithm outputs either (A) or (B), making $O(\frac{\log \delta^{-1}}{\sqrt{\gamma}})$ queries to A .

The procedure of $\text{QAA}(A, \gamma, \delta)$ is presented in Algorithm 1. Here, G , the so-called Grover operator, is defined as

$$G := -AS_0A^{-1}S_\chi. \quad (14)$$

S_χ is an operator that acts as $S_\chi|\psi\rangle|0\rangle = |\psi\rangle|0\rangle$ and $S_\chi|\psi\rangle|1\rangle = -|\psi\rangle|0\rangle$, where $|\psi\rangle$ is any state on the n -qubit register, and implemented just as a Z gate on the single-qubit register. The operator S_0 acts as $S_0|0\rangle|0\rangle = -|0\rangle|0\rangle$ and $S_0|\Psi\rangle = |\Psi\rangle$ for any other computational basis states $|\Psi\rangle$ on the system. We can implement this using a multicontrolled Z gate.

Before presenting the proof of Theorem 2, let us roughly see how Algorithm 1 works. We can show that, for any $j \in \mathbb{N}$,

$$G^j|\Phi\rangle = \sin((2j + 1)\theta_a)|\phi_1\rangle|1\rangle + \cos((2j + 1)\theta_a)|\phi_0\rangle|0\rangle \quad (15)$$

holds, where $\theta_a = \arcsin(\sqrt{a})$ [40]. Therefore, operating G $O(1/\sqrt{a})$ times on $|\Phi\rangle$ makes the amplitude of $|\phi_1\rangle|1\rangle$ of order 1, which means high probability to obtain 1 on the single-qubit register.

Now, the proof of Theorem 2 is as follows.

Proof of Theorem 2. To begin with, note some differences between Algorithm 1 and **QSearch** in [40]. First, in Algorithm 1, loop 5–10⁴ has a bound L on the iteration number, whereas **QSearch** has no bound in the corresponding loop and can run forever. Second, Algorithm 1 repeats state generations and measurements in loop 2–4 and loop 7–9, whereas in **QSearch** they are not repeated.

Under these differences, Algorithm 1 behaves as follows. If $a \geq \frac{3}{4}$, loop 2–4 outputs (A) with probability at least

$$1 - (1 - a)^{m_1} \geq 1 - \left(\frac{1}{4}\right)^{m_1} \geq 1 - \left(\frac{1}{4}\right)^{\log_{\frac{1}{4}} \delta} \geq 1 - \delta. \quad (16)$$

In this, A is called at most $m_1 = O(\log \delta^{-1})$ times, regardless of the value of γ .

On the other hand, if $\gamma \leq a < \frac{3}{4}$, the algorithm works as follows. Loop 7–9 with $l = \tilde{l}(a) := \lceil \log_{\frac{3}{4}} \frac{3}{4\sqrt{a}} \rceil$ outputs (A) with probability at least

$$\begin{aligned} 1 - \left[1 - \frac{1}{2} \left(1 - \frac{1}{2M\sqrt{a}} \right) \right]^{m_2} \\ = 1 - \left(\frac{1}{2} + \frac{1}{4M\sqrt{a}} \right)^{m_2} \end{aligned}$$

³In this ascending sort operation, unitarity holds in the system including ancillary qubits.

⁴Here, loop a - b means that the loop from line a to line b in Algorithm 1.

Algorithm 1. QAA(A, γ, δ), a modified version of **QSearch** in [40] with $c = 3/2$.

Input: A in Eq. (13), G in Eq. (15), $\gamma \in (0, 1)$, $\delta \in (0, 1)$

- 1: Set $L = \lceil \log_{\frac{3}{2}} \frac{3}{4\sqrt{\gamma}} \rceil$, $m_1 = \lceil \log_{\frac{3}{2}} \delta \rceil$, and $m_2 = \lceil \log_{\frac{3}{2}} \delta \rceil$.
- 2: **for** $i = 1$ to m_1 **do**
- 3: Generate $|\Phi\rangle$ and measure the single-qubit register. If the outcome is 1, output (A).
- 4: **end for**
- 5: **for** $l = 1$ to L **do**
- 6: Randomly choose an integer $j \in [M]$, where $M = \lceil (\frac{3}{2})^l \rceil$.
- 7: **for** $i = 1$ to m_2 **do**
- 8: Generate $G^j|\Phi\rangle$ and measure the single-qubit register. If the outcome is 1, output (A).
- 9: **end for**
- 10: **end for**
- 11: Output (B).

$$\begin{aligned} &\geq 1 - \left(\frac{1}{2} + \frac{1}{4\sqrt{a}(\frac{3}{2})^{l(a)}} \right)^{m_2} \\ &\geq 1 - \left(\frac{1}{2} + \frac{1}{3} \right)^{\log_{\frac{3}{2}} \delta} = 1 - \delta, \end{aligned} \tag{17}$$

since, according to [40], one run of line 8 outputs (A) with probability at least $\frac{1}{2}(1 - \frac{1}{2M\sqrt{a}})$ if $0 < a < 3/4$. The number of queries to A until we get (A) is evaluated as follows. Since G contains two calls to A , loop 7–9 with $l = l'$ makes at most $O(m_2(\frac{3}{2})^{l'})$ queries to A for generation of $G^j|\Phi\rangle$. Therefore, until we get (A), A is called $O(\sum_{l=1}^L (\frac{3}{2})^l m_2)$ times, that is, $O(\log \delta^{-1}/\sqrt{a})$ times.

In summary, if $\gamma \leq a \leq 1$, Algorithm 1 outputs (A) with probability at least $1 - \delta$ making $O(\log \delta^{-1}/\sqrt{a})$ queries to A .

To show the statement on the case that $a < \gamma$, we need only to show that the maximum number of queries to A in this algorithm is $O(\log \delta^{-1}/\sqrt{\gamma})$. This is actually true, since the number of queries to A in loop 5–10 is $O(\sum_{l=1}^L (\frac{3}{2})^l m_2)$, that is, $O(\log \delta^{-1}/\sqrt{\gamma})$, and adding $O(\log \delta^{-1})$ queries in loop 2–4 does not change the order.

The statement on qubit number is obvious, since every operation in Algorithm 1 is done by A or G , which is an operator on the system consisting of a n -qubit register and a single qubit register. ■

Let us make some comments on QAA. First, note that QAA can be in fact regarded as an extension of Grover’s algorithm, since the search problem in Theorem 1 can be solved by QAA. This is because we can generate the following state by O_N^{EqPr} and O_F :

$$\frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} |x\rangle |F(x)\rangle = \frac{1}{\sqrt{N}} |x_{\text{tar}}\rangle |1\rangle + \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n \setminus \{x_{\text{tar}}\}} |x\rangle |0\rangle, \tag{18}$$

which is in the form of Eq. (13) with $|\phi_1\rangle = |x_{\text{tar}}\rangle$. Second, QAA provides a quadratic speedup like Grover’s algorithm. Instead of QAA, we can repeat generating $|\Phi\rangle$ and measuring the qubit until we get the measurement outcome 1 and the state $|\phi_1\rangle$. This naive way yields $O(a^{-1})$ repetitions in expectation. Therefore, QAA is quadratically faster than this. Third, note that, in Theorem 2, the marked state is defined as a state in which some qubit takes $|1\rangle$. Although in the original algorithm the marked state can be set more generally [40], the above

setting is sufficient for the proposed algorithm for GW matched filtering, as we will see later.

E. Quantum amplitude estimation

Based on QAA, we can construct an algorithm called QAE for estimating the amplitude of a target state in a superposition state, or, more specifically, a in the state like Eq. (13) [40]. Roughly speaking, in the algorithm, we generate a superposition of states in the form of $G^j|\Phi\rangle$ with various values of j by iteratively operating G controlled by some register R_{QFT} , and outputs an approximation of θ_a onto R_{QFT} by quantum Fourier transform (QFT).⁵ Here, we do not enter details of the procedure but just present the following theorem, which is a modification of Theorem 12 in [40], without proof.

Theorem 3. Suppose that we are given an access to an oracle A in Eq. (13). Then, for any integer t larger than 2, there is an oracle $\tilde{O}_{A,t}^{\text{QAE}}$ that acts as $\tilde{O}_{A,t}^{\text{QAE}}|0\rangle = \sum_{y \in \mathcal{Y}} \alpha_y |y\rangle$, where some ancillary qubits are undisplayed. Here, \mathcal{Y} is a finite set of real numbers that includes a subset $\tilde{\mathcal{Y}}$ consisting of elements \tilde{a} satisfying

$$|\tilde{a} - a| \leq \frac{2\pi\sqrt{a(1-a)}}{t} + \frac{\pi^2}{t^2}, \tag{19}$$

and $\{\alpha_y\}_{y \in \mathcal{Y}}$ are complex numbers satisfying $\sum_{\tilde{y} \in \tilde{\mathcal{Y}}} |\alpha_{\tilde{y}}|^2 \geq 8/\pi^2$. In $\tilde{O}_{A,t}^{\text{QAE}}$, $O_{\mathcal{X}}$ is used $O(t)$ times and $O(n + \log t)$ qubits are used.

Here are some comments. In the original algorithm **Est_Amp** in [40], we measure the state $\sum_{y \in \mathcal{Y}} \alpha_y |y\rangle$ and obtain an estimate on a . However, we now stop the procedure at generation of the state, since, as explained below, we use QAE as a subroutine for the SNR calculation in searching high SNR templates by QAA and thus require it to be a unitary operation. Besides, note that the statement on qubit number is obvious since **Est_Amp** in [40] uses only the register R_{QFT} , which has $O(\log t)$ qubits, along with the system on which A acts.

⁵There are some variants of QAE that rely on not QFT but iterative measurements and processing outcomes [41–50]. However, we do not use these in this paper since, in the proposed algorithm, we require QAE to be a unitary operation as a subroutine in QAA, as mentioned below.

We often want to enhance the lower bound $8/\pi^2$ on the success probability of QAE to a given high value. We can accomplish this thanks to the following theorem, which is Lemma 1 in [15] and originally Lemma 6.1 in [51].

Theorem 4. Let $\mu \in \mathbb{R}$ and $\epsilon \in \mathbb{R}_+$. Let \mathcal{A} be an algorithm that outputs an ϵ approximation of μ with probability $\gamma \geq \frac{3}{4}$. Then, for any $\delta \in (0, 1)$, the median of outputs in $12 \lceil \log \delta^{-1} \rceil + 1$ runs of \mathcal{A} is an ϵ approximation of μ with probability at least $1 - \delta$.

This implies the following. Letting N be an integer larger than $12 \lceil \log \delta^{-1} \rceil + 1$, we generate the state

$$|\Psi_N\rangle := \sum_{y_1, \dots, y_N \in \mathcal{Y}} \alpha_{y_1} \cdots \alpha_{y_N} |y_1\rangle \cdots |y_N\rangle |\text{med}(y_1, \dots, y_N)\rangle \quad (20)$$

by operating $\tilde{O}_{A,t}^{\text{QAE}}$ on each of the first N registers and using O_N^{med} . We rewrite this state as $|\Psi_N\rangle := \alpha_{\leq \epsilon} |\psi_{\leq \epsilon}\rangle + \alpha_{> \epsilon} |\psi_{> \epsilon}\rangle$ with $\alpha_{\leq \epsilon}, \alpha_{> \epsilon} \in \mathbb{C}$ and the states

$$\begin{aligned} |\psi_{\leq \epsilon}\rangle &:= \sum_{\substack{\tilde{a} \in \mathcal{Z} \\ |\tilde{a}-a| \leq \frac{2\pi\sqrt{a(1-a)}}{t} + \frac{\pi^2}{t^2}}} \beta_{\tilde{a}} |\phi_{\tilde{a}}\rangle |\tilde{a}\rangle, \\ |\psi_{> \epsilon}\rangle &:= \sum_{\substack{\tilde{a} \in \mathcal{Z} \\ |\tilde{a}-a| > \frac{2\pi\sqrt{a(1-a)}}{t} + \frac{\pi^2}{t^2}}} \beta_{\tilde{a}} |\phi_{\tilde{a}}\rangle |\tilde{a}\rangle, \end{aligned} \quad (21)$$

where \mathcal{Z} is some finite set of real numbers, $\{\beta_{\tilde{a}}\}_{\tilde{a} \in \mathcal{Z}}$ are complex numbers, and $\{|\phi_{\tilde{a}}\rangle\}_{\tilde{a} \in \mathcal{Z}}$ are states on the first N registers. Then, $|\alpha_{\leq \epsilon}|^2$, the squared amplitude of the state $|\psi_{\leq \epsilon}\rangle$, in which the number \tilde{a} on the last register satisfies Eq. (19), is larger than $1 - \delta$. This technique is used in the quantum algorithm for Monte Carlo integration, which is explained next.

F. Quantum Monte Carlo integration

On the basis of QAE, we can construct a quantum algorithm for Monte Carlo integration [15], which we hereafter call QMCI. Although Monte Carlo integration is generally a method to estimate integrals, we now consider it as a method to estimate the mean of \mathcal{X} a given set of real numbers, since it is sufficient for the proposed algorithm. Among some versions presented in [15], we use the one for the situation where an upper bound on $\text{Var}(\mathcal{X})$ is given.

Theorem 5. Let $N \in \mathbb{N}$ and \mathcal{X} be a set of N real numbers, X_0, \dots, X_{N-1} , whose mean is $\mu := \frac{1}{N} \sum_{i=0}^{N-1} X_i$ and sample variance $\text{Var}(\mathcal{X})$ satisfies $\text{Var}(\mathcal{X}) \leq \sigma^2$ with $\sigma \in \mathbb{R}_+$. Suppose that we are given an oracle $O_{\mathcal{X}}$ that acts on a system with $O(\log N)$ qubits in total as

$$O_{\mathcal{X}} |i\rangle |0\rangle = |i\rangle |X_i\rangle, \quad (22)$$

for any $i \in [N]_0$. Let $\epsilon \in (0, 4\sigma)$ and $\delta \in (0, 1)$. Then, there is an oracle $O_{\mathcal{X}, \epsilon, \delta, \sigma}^{\text{mean}}$ such that

$$O_{\mathcal{X}, \epsilon, \delta, \sigma}^{\text{mean}} |0\rangle = \sum_{y \in \mathcal{Y}} \alpha_y |y\rangle, \quad (23)$$

where some ancillary qubits are undisplayed. Here, \mathcal{Y} is a finite set of real numbers that includes a subset $\tilde{\mathcal{Y}}$ consisting

of ϵ approximations of μ and $\{\alpha_y\}_{y \in \mathcal{Y}}$ are complex numbers satisfying $\sum_{y \in \tilde{\mathcal{Y}}} |\alpha_y|^2 \geq 1 - \delta$. In $O_{\mathcal{X}, \epsilon, \delta, \sigma}^{\text{mean}}$,

$$O \left[\frac{\sigma}{\epsilon} \log^{3/2} \left(\frac{\sigma}{\epsilon} \right) \log \log \left(\frac{\sigma}{\epsilon} \right) \log \left(\frac{1}{\delta} \right) \right] \quad (24)$$

queries to $O_{\mathcal{X}}$ are made and

$$O \left\{ \left[\log N + \log \left(\frac{\sigma}{\epsilon} \right) \right] \log \left(\frac{\sigma}{\epsilon} \right) \log \log \left(\frac{\sigma}{\epsilon} \right) \log \delta^{-1} \right\} \quad (25)$$

qubits are used.

We present the proof in Appendix A.

III. QUANTUM ALGORITHM FOR GRAVITATIONAL WAVE MATCHED FILTERING

A. Problem and assumptions

Equipped with the above quantum algorithms, we now consider applying them to GW matched filtering. We start from formally stating the problem we consider.

Problem 1. Let T, M , and N_{temp} be a positive real number, a positive even integer and a positive integer, respectively. Define $\Delta t := T/M$ and, for $k \in [M/2 - 1]$, $f_k := k/T$. Suppose that we are given a complex sequence $\{\tilde{s}(f_k)\}_{k \in [M/2 - 1]}$, a function $S_n : \mathbb{R}_+ \rightarrow \mathbb{R}_+$, and, for every $m \in [N_{\text{temp}}]_0$, a function $\tilde{Q}_m : \mathbb{R}_+ \rightarrow \mathbb{C}$. Then, determine whether there exists any $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$ such that $\rho_{m,j}$ in Eq. (11) exceeds some given value or not. If there are such integer pairs, find one of them.

We need some preparations to tackle this problem. First, let us define some quantities for convenience. The first one is as follows: for $\rho \in \mathbb{R}_+$,

$$\tilde{r}(\rho) := \frac{\#\{(m, j) \in [N_{\text{temp}}]_0 \times [M]_0 \mid \rho_{m,j} \geq \rho\}}{N_{\text{temp}} M}. \quad (26)$$

Obviously, $\tilde{r}(\rho)$ represents the fraction of templates that give SNRs larger than ρ and is an analog of $r(\rho)$ in Eq. (9). The next one is about the magnitude of template functions \tilde{Q}_m normalized by S_n ,

$$\gamma := \max_{(m,k) \in [N_{\text{temp}}]_0 \times [\frac{M}{2} - 1]} \frac{|\tilde{Q}_m(f_k)|}{\sqrt{S_n(f_k) \Delta t}}. \quad (27)$$

We will see in Theorem 6 that the large γ leads to the large complexity in our algorithm, and also see in Sec. III D 4 that it is not so large that the quadratic quantum speedup by our algorithm is ruined.

Next, let us make some assumptions needed to discuss the quantum algorithm and its complexity. The first one is about availability of some fundamental oracles.

Assumption 1. We have accesses to oracles O_{Re} and O_{Im} such that, for every $(m, k) \in [N_{\text{temp}}]_0 \times [\frac{M}{2}]_0$,

$$\begin{aligned} O_{\text{Re}} |m\rangle |k\rangle |0\rangle &= \begin{cases} |m\rangle |k\rangle |0\rangle; & \text{if } k = 0 \\ |m\rangle |k\rangle |\Re(\frac{2\tilde{Q}_m^*(f_k)\tilde{s}(f_k)}{S_n(f_k)\Delta t})\rangle; & \text{otherwise} \end{cases}, \\ O_{\text{Im}} |m\rangle |k\rangle |0\rangle &= \begin{cases} |m\rangle |k\rangle |0\rangle; & \text{if } k = 0 \\ |m\rangle |k\rangle |\Im(\frac{2\tilde{Q}_m^*(f_k)\tilde{s}(f_k)}{S_n(f_k)\Delta t})\rangle; & \text{otherwise} \end{cases} \end{aligned} \quad (28)$$

Algorithm 2. Previous algorithm for GW matched filtering (modified).

Input:

- $\delta \in (0, 1)$.
- $\rho_{th} \in \mathbb{R}_+$ the SNR threshold.
- An oracle O_{FFT} such that, for every $m \in [N_{temp}]_0$,

$$O_{FFT}|m\rangle|0\rangle = |m\rangle|\rho_m\rangle. \tag{30}$$

1: Combining $O_{N_{temp}}^{EqPr}$, O_{FFT} and a comparer, construct an oracle O'_{FFT} that acts as

$$O'_{FFT}|0\rangle|0\rangle|0\rangle = \frac{1}{\sqrt{N_{temp}}} \sum_{m=0}^{N_{temp}-1} |m\rangle|\rho_m\rangle|\mathbb{1}_{\rho_m \geq \rho_{th}}\rangle. \tag{31}$$

- 2: Run QAA(O'_{FFT} , $\frac{1}{N_{temp}}$, δ).
- 3: **if** we get the message “failure” **then**
- 4: Output the message “there is no signal”.
- 5: **else**
- 6: Measure the first register in the quantum state output by QAA and let the outcome be m .
- 7: Calculate ρ_m classically by FFT.
- 8: **if** $\rho_m \geq \rho_{th}$ **then**
- 9: Output the message “there is a signal” and m .
- 10: **else**
- 11: Output the message “there is no signal”.
- 12: **end if**
- 13: **end if**

We will discuss how to implement these in Sec. III D 2. The next assumption is on the mean and the variance of detector outputs used in the SNR calculation.

Assumption 2.

$$\begin{aligned} \left\{ \text{Mean} \left[\left(\frac{2\Re\tilde{s}(f_k)}{\sqrt{S_n(f_k)T}} \right)_{k=1, \dots, M/2-1} \right] \right\}^2 &\leq 1, \\ \left\{ \text{Mean} \left[\left(\frac{2\Im\tilde{s}(f_k)}{\sqrt{S_n(f_k)T}} \right)_{k=1, \dots, M/2-1} \right] \right\}^2 &\leq 1, \\ \text{Var} \left[\left(\frac{2\Re\tilde{s}(f_k)}{\sqrt{S_n(f_k)T}} \right)_{k=1, \dots, M/2-1} \right] &\leq 4, \\ \text{Var} \left[\left(\frac{2\Im\tilde{s}(f_k)}{\sqrt{S_n(f_k)T}} \right)_{k=1, \dots, M/2-1} \right] &\leq 4. \end{aligned} \tag{29}$$

We will discuss the validity of this in Sec. III D 3. As we will see in Sec. III C, this assumption is important for the proposed quantum algorithm, since it leads to the upper bound on the variance of summands in the SNR calculation, with which we can use QMCI for variables with bounded variance.

B. Previous algorithm

Before the quantum algorithm, we review the algorithm proposed in [3]. It is shown as Algorithm 2, which is a modified version of Algorithm 2 in [3]. Given a SNR threshold ρ_{th} , this algorithm outputs the message “there is a signal” and $m \in [N_{temp}]_0$ such that $\rho_m \geq \rho_{th}$ with probability at least $1 - \delta$, if there exists such m . Note that Algorithm 2 in our paper uses QAA instead of QAE on $O'_{FFT}|0\rangle|0\rangle|0\rangle$ in Algorithm 2 of [3], which does not affect the scaling of the complexity on M and N_{temp} . Also note that we can implement O_{FFT} in Eq. (30)

by arithmetic oracles, since FFT is actually a sequence of arithmetic operations.⁶

The number of queries to arithmetic oracles in this algorithm is⁷

$$O\left(\frac{M \log M}{\sqrt{r(\rho_{th})}}\right), \tag{32}$$

when $r(\rho_{th}) > 0$ and

$$O(\sqrt{N_{temp}}M \log M) \tag{33}$$

when $r(\rho_{th}) = 0$. We can see this as follows. QAA makes $O(1/\sqrt{r(\rho_{th})})$ calls to O'_{FFT} and thus to O_{FFT} when $r(\rho_{th}) > 0$, and $O(\sqrt{N_{temp}})$ calls to them when $r(\rho_{th}) = 0$. Besides, the number of queries to arithmetic oracles in O_{FFT} is of order $O(M \log M)$ like the number of floating-point operations in FFT on a classical computer. Combining these, we get the complexity bounds in Eqs. (32) and (33). They show a quadratic speedup over the classical complexity of Eqs. (8) and (10) with respect to $1/r(\rho_{th})$ and N_{temp} .

When it comes to qubit number, Algorithm 2 uses $O(M)$ qubits, since FFT calculates $\rho_{m,0}, \dots, \rho_{m,M-1}$ simultaneously and thus use $O(M)$ registers to store intermediate and final calculation results.

⁶In fact, implementation of FFT as a quantum circuit has been studied in [52].

⁷Although Eqs. (32) and (33) do not have a factor $\log N_{temp}$, while Eq. (35) in [3] has, we omit this reasonably assuming that $M \log M \gg \log N_{temp}$.

C. Proposed algorithm and its complexity

1. Idea

The previous algorithm for GW matched filtering explained above uses FFT for the SNR calculation. On the other hand, from the formula Eq. (11) for SNR, we conceive the following idea: can we use QMCI for the SNR calculation? As we will see later, we can construct an oracle O_ρ to calculate the summand $\tilde{\rho}_{m,j,k}$ in the SNR calculation in Eq. (11) making $O(1)$ uses of O_{Re} and O_{Im} , and thus apply QMCI following Theorem 5. If we can set a bound σ^2 on the sample variance of $\{\tilde{\rho}_{m,j,k}\}_{k=1,\dots,\frac{M}{2}-1}$ and the accuracy ϵ in the SNR calculation, the query complexity with respect to O_{Re} and O_{Im} is of order $\tilde{O}(\sigma/\epsilon)$.

This QMCI-based approach has a benefit on the qubit number reduction. As shown in Eq. (25), the number of qubits QMCI uses scales on M as $O(\text{polylog}M)$, since the SNR given as Eq. (11) is a mean of $O(M)$ terms. This means large reduction compared to the number required by FFT, which is of order $O(M)$. This provides a large benefit, since quantum computers will have a limitation on qubit capacity even in the future as mentioned in Introduction.

When it comes to the query complexity, the QMCI-based method is roughly same as the previous FFT-based algorithm. One might concern that the proposed method might worsen the scaling on M due to the expansion of the space searched by QAA. Unlike FFT, which simultaneously calculates $\rho_{m,j}$ for all $j \in [M]_0$, QMCI is performed for each j . Thus, the search in the parameter space of $m \in [N_{\text{temp}}]_0$ to find a large ρ_m in the previous algorithm is replaced with the search in a larger parameter space of $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$ to find a large $\rho_{m,j}$. This means that the iteration number in QAA increases from $\tilde{O}(1/\sqrt{r(\rho_{\text{th}})})$ to $\tilde{O}(1/\sqrt{\tilde{r}(\rho_{\text{th}})})$, which is $\tilde{O}(\sigma\sqrt{M}/\epsilon\sqrt{r(\rho_{\text{th}})})$ at maximum since $\tilde{r}(\rho_{\text{th}}) > r(\rho_{\text{th}})/M$. Combining the aforementioned query complexity of the SNR calculation by QMCI, we bound the total number of queries to O_{Re} and O_{Im} as $\tilde{O}(\sigma\sqrt{M}/\epsilon\sqrt{r(\rho_{\text{th}})})$. Fortunately, since we can set $\sigma = O(\sqrt{M})$ and $\epsilon = O(1)$ as we will see below, the query complexity is resultingly $\tilde{O}(M/\sqrt{r(\rho_{\text{th}})})$, which is same as the number of queries to arithmetic oracles in the FFT-based method. Note that this is a reasonable comparison, since O_{Re} and O_{Im} are constructed by arithmetic oracles, with the aid of the quantum random access memory (QRAM) [53], as explained later.

We should also note that the nature of QMCI causes the following issue. The output of QMCI inevitably accompanies an error, and thus, even if the SNR of a given template calculated by QMCI exceeds the threshold ρ_{th} , its true SNR might be below ρ_{th} . We may think that we can evade such a misjudge by setting the accuracy ϵ in QMCI extremely small, but it comes with large complexity. Therefore, we need to reasonably set the accuracy: following the nature of the problem under consideration, we should derive the error tolerance for SNR and set the QMCI accuracy matching it.

For this, we propose the following way. We consider GW matched filtering as a system that alarms us when the detector output seems to contain a signal. Besides, we consider the two levels of SNR threshold denoted by ρ_{hard} and ρ_{soft} , which have the following meaning.

- (i) If some templates have SNR $\rho \geq \rho_{\text{hard}}$ for a given detector output, we want to be alarmed with certainty.
- (ii) We never want to be falsely alarmed when all templates have SNR $\rho < \rho_{\text{soft}}$.
- (iii) When no template has SNR $\rho \geq \rho_{\text{hard}}$ but some have $\rho \in [\rho_{\text{soft}}, \rho_{\text{hard}})$, it is not needed but fine to be alarmed.

In this situation, we set the QMCI accuracy to $(\rho_{\text{hard}} - \rho_{\text{soft}})/2$ and judge a template as matched if its SNR calculated by QMCI exceeds $(\rho_{\text{hard}} + \rho_{\text{soft}})/2$ and mismatched otherwise. In this strategy, with high probability, a template with a true SNR $\rho \geq \rho_{\text{hard}}$ is judged as matched and that with a true SNR $\rho < \rho_{\text{soft}}$ is judged as mismatched. A template with a true SNR $\rho_{\text{soft}} \leq \rho < \rho_{\text{hard}}$ is an intermediate case where the data may contain a signal near the threshold and can be judged as either matched or mismatched due to the QMCI error. We present an illustration of this strategy in Fig. 1. We discuss how to set the two levels ρ_{hard} and ρ_{soft} in Sec. III D 1.

Note that “match” and “mismatch” discussed here are the result of quantum computation for a given detector output, and do not necessarily mean the result correctly indicates whether there is a GW signal or not. In fact, a false detection can occur due to the random instrumental noise, which is not taken into account here. It is, on the other hand, relevant to how we set the thresholds of ρ_{hard} and ρ_{soft} . See Sec. III D 1 for more detailed discussion.

Based on this idea, we design a quantum algorithm for GW matched filtering in the following part.

2. Supporting lemma on the variance of summands in the SNR calculation

Here, as a preparation to present the algorithm, let us prove the following lemma on the variance of summands in the SNR calculation, which follows from Assumption 2.

Lemma 1. Under Assumption 2,

$$\text{Var}(\{\tilde{\rho}_{m,j,k}\}_{k=0,\dots,M/2-1}) \leq 80M\gamma^2 \tag{34}$$

holds for every $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$.

Proof. We can write $\tilde{\rho}_{m,j,k} = \tilde{\rho}_{m,j,k}^{(1)} + \tilde{\rho}_{m,j,k}^{(2)} + \tilde{\rho}_{m,j,k}^{(3)} + \tilde{\rho}_{m,j,k}^{(4)}$ with

$$\begin{aligned} \tilde{\rho}_{m,j,k}^{(1)} &= \sqrt{M} \Re \left(\frac{\tilde{Q}_m^*(f_k)}{\sqrt{S_n(f_k)\Delta t}} \right) \frac{2\Re(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}} \cos \left(\frac{2\pi jk}{M} \right), \\ \tilde{\rho}_{m,j,k}^{(2)} &= -\sqrt{M} \Im \left(\frac{\tilde{Q}_m^*(f_k)}{\sqrt{S_n(f_k)\Delta t}} \right) \frac{2\Im(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}} \cos \left(\frac{2\pi jk}{M} \right), \end{aligned}$$

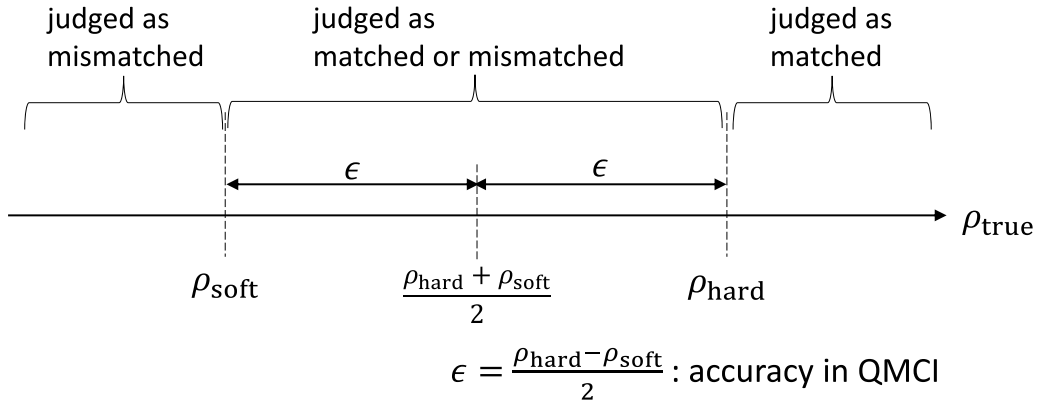


FIG. 1. An illustration of the current strategy. For every template, if the SNR calculated by QMCI exceeds $(\rho_{\text{hard}} + \rho_{\text{soft}})/2$, we judge it as matched, and otherwise we judge it as mismatched. With QMCI accuracy $(\rho_{\text{hard}} - \rho_{\text{soft}})/2$, this leads to a correct judgment for templates with a true SNR $\rho_{\text{true}} < \rho_{\text{soft}}$ and $\rho_{\text{true}} \geq \rho_{\text{hard}}$ with high probability.

$$\begin{aligned}\tilde{\rho}_{m,j,k}^{(3)} &= -\sqrt{M}\Re\left(\frac{\tilde{Q}_m^*(f_k)}{\sqrt{S_n(f_k)\Delta t}}\right)\frac{2\Im(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}}\sin\left(\frac{2\pi jk}{M}\right), \\ \tilde{\rho}_{m,j,k}^{(4)} &= -\sqrt{M}\Im\left(\frac{\tilde{Q}_m^*(f_k)}{\sqrt{S_n(f_k)\Delta t}}\right)\frac{2\Re(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}}\sin\left(\frac{2\pi jk}{M}\right).\end{aligned}\quad (35)$$

We see that

$$\begin{aligned}\text{Var}(\{\tilde{\rho}_{m,j,k}^{(1)}\}_{k=0,\dots,M/2-1}) &\leq \text{Mean}(\{(\tilde{\rho}_{m,j,k}^{(1)})^2\}_{k=0,\dots,M/2-1}) \leq M\gamma^2 \text{Mean}\left(\left\{\left(\frac{2\Re(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}}\right)^2\right\}_{k=1,\dots,M/2-1}\right) \\ &= M\gamma^2 \left[\text{Var}\left(\left\{\frac{2\Re(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}}\right\}_{k=1,\dots,M/2-1}\right) + \left(\text{Mean}\left(\left\{\frac{2\Re(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}}\right\}_{k=1,\dots,M/2-1}\right)\right)^2 \right] \leq 5M\gamma^2,\end{aligned}\quad (36)$$

where the second inequality follows from

$$(\tilde{\rho}_{m,j,k}^{(1)})^2 = M \left[\Re\left(\frac{\tilde{Q}_m^*(f_k)}{\sqrt{S_n(f_k)\Delta t}}\right) \right]^2 \cos^2\left(\frac{2\pi jk}{M}\right) \left(\frac{2\Re(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}}\right)^2 \leq M\gamma^2 \left(\frac{2\Re(\tilde{s}(f_k))}{\sqrt{S_n(f_k)T}}\right)^2,\quad (37)$$

for $k \in [\frac{M}{2} - 1]$ and $\tilde{\rho}_{m,j,0}^{(1)} = 0$, and the last inequality follows from Assumption 2. Similarly, we have $\text{Var}(\{\tilde{\rho}_{m,j,k}^{(2)}\}_{k=0,\dots,M/2-1}) \leq 5M\gamma^2$, $\text{Var}(\{\tilde{\rho}_{m,j,k}^{(3)}\}_{k=0,\dots,M/2-1}) \leq 5M\gamma^2$ and $\text{Var}(\{\tilde{\rho}_{m,j,k}^{(4)}\}_{k=0,\dots,M/2-1}) \leq 5M\gamma^2$. Combining these with

$$\begin{aligned}\text{Var}(\{\tilde{\rho}_{m,j,k}\}_{k=0,\dots,M/2-1}) &= \sum_{a=1}^4 \text{Var}(\{\tilde{\rho}_{m,j,k}^{(a)}\}_{k=0,\dots,M/2-1}) + \sum_{\substack{a,b=1,\dots,4 \\ a \neq b}} \text{Cov}(\{\tilde{\rho}_{m,j,k}^{(a)}\}_{k=0,\dots,M/2-1}, \{\tilde{\rho}_{m,j,k}^{(b)}\}_{k=0,\dots,M/2-1}) \\ &\leq \sum_{a=1}^4 \text{Var}(\{\tilde{\rho}_{m,j,k}^{(a)}\}_{k=0,\dots,M/2-1}) + \sum_{\substack{a,b=1,\dots,4 \\ a \neq b}} \sqrt{\text{Var}(\{\tilde{\rho}_{m,j,k}^{(a)}\}_{k=0,\dots,M/2-1}) \text{Var}(\{\tilde{\rho}_{m,j,k}^{(b)}\}_{k=0,\dots,M/2-1})},\end{aligned}\quad (38)$$

we obtain Eq. (34). ■

3. Main result

Then, the following is our main result, a quantum algorithm for GW matched filtering and a theorem on its query complexity and the number of qubits used.

Theorem 6. Under Assumptions 1 and 2, consider Problem 1. Let ρ_{soft} and ρ_{hard} be real numbers such that $0 < \frac{\rho_{\text{hard}} - \rho_{\text{soft}}}{8\sqrt{5}M\gamma} < 4$, and δ be a real number in $(0, 1)$. Then, there is a quantum algorithm that uses

$$O\left\{\left[\log M + \log\left(\frac{\sqrt{M}\gamma}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right)\right] \log\left(\frac{\sqrt{M}\gamma}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log\log\left(\frac{\sqrt{M}\gamma}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log\left(\frac{N_{\text{temp}}M}{\delta}\right)\right\}\quad (39)$$

qubits and behaves as follows:

- (i) The algorithm outputs either of

- (A) a message “there is a signal” and an integer pair $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$ such that $\rho_{m,j} \geq \rho_{\text{soft}}$,
 (B) a message “there is no signal”.

(ii) If $r(\rho_{\text{hard}}) > 0$, the algorithm outputs (A) with probability at least $1 - \delta$. In the algorithm, the number of queries to O_{Re} and O_{Im} is of order

$$O\left[\frac{\gamma\sqrt{M}}{(\rho_{\text{hard}} - \rho_{\text{soft}})\sqrt{r(\rho_{\text{hard}})}} \log^{3/2}\left(\frac{\gamma\sqrt{M}}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log \log\left(\frac{\gamma\sqrt{M}}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log\left(\frac{N_{\text{temp}}M}{\delta}\right) \log \delta^{-1}\right], \quad (40)$$

and thus

$$O\left[\frac{\gamma M}{(\rho_{\text{hard}} - \rho_{\text{soft}})\sqrt{r(\rho_{\text{hard}})}} \log^{3/2}\left(\frac{\gamma\sqrt{M}}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log \log\left(\frac{\gamma\sqrt{M}}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log\left(\frac{N_{\text{temp}}M}{\delta}\right) \log \delta^{-1}\right]. \quad (41)$$

(iii) If $r(\rho_{\text{soft}}) = 0$, the algorithm outputs (B) with certainty. In the algorithm, the number of queries to O_{Re} and O_{Im} is of order

$$O\left[\frac{\gamma M \sqrt{N_{\text{temp}}}}{\rho_{\text{hard}} - \rho_{\text{soft}}} \log^{3/2}\left(\frac{\gamma\sqrt{M}}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log \log\left(\frac{\gamma\sqrt{M}}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log\left(\frac{N_{\text{temp}}M}{\delta}\right) \log \delta^{-1}\right]. \quad (42)$$

(iv) If $r(\rho_{\text{hard}}) = 0$ and $r(\rho_{\text{soft}}) > 0$, the algorithm outputs either (A) or (B). In the algorithm, the number of queries to O_{Re} and O_{Im} is of order as in Eq. (42).

Proof. We first present the algorithm, and then prove the statements on the query complexity and the qubit number.

Algorithm

The algorithm is shown in Algorithm 3. Note that, because of Lemma 1, we can set σ , the upper bound on the variance of $\{\tilde{\rho}_{m,j,k}\}_{k=0,\dots,M/2-1}$, as in line 1.

Query complexity and qubit number

We describe this part in B since it is rather technical. ■

On the probability that we obtain the output (A) in the case that $r(\rho_{\text{hard}}) = 0$ and $r(\rho_{\text{soft}}) > 0$, we now give not a rigorous analysis but a qualitative comment. That probability is affected by the dependence of $|\alpha_{\rho,y}|^2$ on y , where $\alpha_{\rho,y}$ appears in Eq. (44). In other words, it is affected by the probability distribution of the output of $O_{\rho,\epsilon,\delta',\sigma}^{\text{mean}}$, the oracle to compute the SNR by QMCI. $O_{\rho,\epsilon,\delta',\sigma}^{\text{mean}}$ is designed so that $\sum_{\tilde{y} \in \tilde{Y}_{m,j}} |\alpha_{\rho,\tilde{y}}|^2 \geq 1 - \delta$, which means that the output is distributed almost within the ϵ neighborhood of the true SNR value $\rho_{m,j}$. Hence, for $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$ such that $\rho_{m,j} \geq \rho_{\text{hard}}$, it is almost certain that the output of $O_{\rho,\epsilon,\delta',\sigma}^{\text{mean}}$ exceeds ρ_{mid} and such (m, j) is judged as “matched”, which means that such (m, j) can be found by QAA in Algorithm 3. On the other hand, as $\rho_{m,j}$ decreases beyond ρ_{hard} , the probability of such (m, j) being judged as “matched” decreases too and becomes almost 0 when $\rho_{m,j} < \rho_{\text{soft}}$. From this observation, we see the following. If $r(\rho_{\text{hard}}) > 0$, Algorithm 3 outputs (A) with probability at least $1 - \delta$, as Theorem 6 states. As $\max_{m,j} \rho_{m,j}$ decreases from ρ_{hard} to ρ_{soft} , the probability of the output (A) decreases to 0. For more detailed analysis on this change of the probability, we need to know the dependence of $|\alpha_{\rho,y}|^2$ on y concretely, but it is difficult since, although the probability distribution of the output of a single QAE is known [40], the QMCI we are using now is the combination of the multiple QAEs (see A) and thus it is hard to analytically obtain the probability distribution of its output.

We also comment that, when $r(\rho_{\text{hard}}) > 0$ or $r(\rho_{\text{hard}}) = 0, r(\rho_{\text{soft}}) > 0$, repeating Algorithm 3 enhance the probability to obtain the output (A): if that probability in a single run of Algorithm 3 is p_A , the probability that we obtain (A) at least once in n runs is $1 - (1 - p_A)^n$.

D. Remarks on settings and assumptions

Now, we discuss the validity on the settings and assumptions in the algorithm proposed above.

1. Setting of two thresholds

In Algorithm 3, we set the two SNR thresholds, ρ_{hard} and ρ_{soft} , whose meanings are explained in Sec. III C 1. We consider the following is a plausible way to set them. First, let us denote a common value of the SNR threshold as ρ_{com} , e.g., $\rho_{\text{com}} = 8^8$ [8]. Note that the calculated SNR, in

general, has fluctuations due to the random detector noise, and its variance is 1 under the current normalization of the templates. Thus, even without the QMCI error, events with $\rho \in [\rho_{\text{com}}, \rho_{\text{com}} + 1)$ could have a true SNR value smaller than the threshold, while events with $\rho > \rho_{\text{com}} + 1$ are very

time Δt_{start} is determined by the mismatch between two waveforms having slightly different coalescence times. If we set the mismatch is 5% and the waveforms is monochromatic with the frequency of 100 Hz, we get $\Delta t_{\text{start}} \sim O(10^{-3})$ sec. Then, for the observation period of T_{obs} , the expected number of false alarm events is estimated by $N_{\text{fa}} \sim p_{\text{fa,temp}} N_{\text{temp}} T_{\text{obs}} / \Delta t_{\text{start}}$. Assuming $T_{\text{obs}} = 10^7$ sec, $\Delta t_{\text{start}} = 10^{-3}$ sec, and $N_{\text{temp}} = 10^6$, we should set the SNR threshold to $\rho_{\text{com}} \sim \sqrt{2 \ln(10^{10} \times N_{\text{temp}})} \sim 8.6$ if we suppress N_{fa} to $O(1)$. See Ref. [54] and Chap.7 of Ref. [55].

⁸In the absence of the astrophysical signal, SNR follows the Rayleigh distribution. The false alarm probability of each template is given by $p_{\text{fa,temp}} = \exp[-\rho_{\text{com}}^2/2]$. The resolution of the coalescence

Algorithm 3. Proposed algorithm for GW matched filtering.

- 1: Set $\epsilon = \frac{\rho_{\text{hard}} - \rho_{\text{soft}}}{2}$, $\sigma = 4\sqrt{5M}\gamma$ and $\delta' = \frac{\delta}{4N_{\text{temp}}M}$.
- 2: Combining O_{Re} , O_{Im} and some arithmetic oracles, construct an oracles O_ρ such that, for every $(m, j, k) \in [N_{\text{temp}}]_0 \times [M]_0 \times [\frac{M}{2}]_0$,

$$O_\rho |m\rangle|j\rangle|k\rangle|0\rangle = |m\rangle|j\rangle|k\rangle|\tilde{\rho}_{m,j,k}\rangle. \tag{43}$$

- 3: On the basis of Theorem 5, using O_ρ , construct an oracle $O_{\rho,\epsilon,\delta',\sigma}^{\text{mean}}$ such that, for every $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$,

$$O_{\rho,\epsilon,\delta',\sigma}^{\text{mean}} |m\rangle|j\rangle|0\rangle = |m\rangle|j\rangle \sum_{y \in \mathcal{Y}_{m,j}} \alpha_{\rho,y}|y\rangle, \tag{44}$$

where $\mathcal{Y}_{m,j}$ is a finite set of real numbers that includes a subset $\tilde{\mathcal{Y}}_{m,j}$ consisting of ϵ approximations of $\rho_{m,j}$ and $\{\alpha_y\}_{y \in \mathcal{Y}_{m,j}}$ are complex numbers satisfying $\sum_{y \in \tilde{\mathcal{Y}}_{m,j}} |\alpha_{\rho,y}|^2 \geq 1 - \delta$.

- 4: Construct an oracle O_{AE} that performs the following operation

$$\begin{aligned} |0\rangle|0\rangle|0\rangle|0\rangle &\rightarrow \sqrt{\frac{1}{N_{\text{temp}}M}} \sum_{m=0}^{N_{\text{temp}}-1} \sum_{j=0}^{M-1} |m\rangle|j\rangle|0\rangle|0\rangle \\ &\rightarrow \sqrt{\frac{1}{N_{\text{temp}}M}} \sum_{m=0}^{N_{\text{temp}}-1} \sum_{j=0}^{M-1} |m\rangle|j\rangle \sum_{y \in \mathcal{Y}_{m,j}} \alpha_{\rho,y}|y\rangle|0\rangle \\ &\rightarrow \sqrt{\frac{1}{N_{\text{temp}}M}} \sum_{m=0}^{N_{\text{temp}}-1} \sum_{j=0}^{M-1} |m\rangle|j\rangle \sum_{y \in \mathcal{Y}_{m,j}} \alpha_{\rho,y}|y\rangle (\mathbb{1}_{y \geq \rho_{\text{mid}}} |1\rangle + \mathbb{1}_{y < \rho_{\text{mid}}} |0\rangle), \end{aligned} \tag{45}$$

where $\rho_{\text{mid}} := \frac{\rho_{\text{hard}} + \rho_{\text{soft}}}{2}$. In Eq. (45), we use $O_{N_{\text{temp}}}^{\text{EqPr}}$ and O_M^{EqPr} at the first arrow, $O_{\rho,\epsilon,\delta',\sigma}^{\text{mean}}$ at the second arrow, and a comparer with $|\rho_{\text{mid}}\rangle$ on an undisplayed ancillary register at the last arrow.

- 5: Run $\text{QAA}(O_{\text{AE}}, \frac{1}{2N_{\text{temp}}M}, \frac{\delta}{2})$.
- 6: **if** we get the message “failure” **then**
- 7: Output the message “there is no signal”.
- 8: **else**
- 9: Measure the first two registers in the quantum state output by QAA and let the outcome be (m, j) .
- 10: Calculate $\rho_{m,j}$ classically.
- 11: **if** $\rho_{m,j} \geq \rho_{\text{soft}}$ **then**
- 12: Output the message “there is a signal” and (m, j) .
- 13: **else**
- 14: Output the message “there is no signal”.
- 15: **end if**
- 16: **end if**

likely to exceed the threshold. In light of this, it is reasonable to set $\rho_{\text{soft}} = \rho_{\text{com}}$ and $\rho_{\text{hard}} = \rho_{\text{com}} + 1$, that is, $\rho_{\text{soft}} = 8$ and $\rho_{\text{hard}} = 9$ for $\rho_{\text{com}} = 8$. In this setting, Algorithm 3 detects events with $\rho \geq 9$ with high probability and never falsely alarms us for events with $\rho < 8$, and events with $\rho \in [8, 9)$ are detected or missed depending on fluctuations by the detector noise and the QMCI error.

2. Implementation of O_{Re} and O_{Im}

Here, we discuss the validity of Assumption 1, that is, implementability of O_{Re} and O_{Im} . If we have accesses to the following oracles O_{hSRe} , O_{hSIm} , O_{QRe} , and O_{QIm} such that, for every $(m, k) \in [N_{\text{temp}}]_0 \times [\frac{M}{2} - 1]$,

$$\begin{aligned} O_{\text{hSRe}} |k\rangle|0\rangle &= |k\rangle \left| \Re \left(\frac{\tilde{h}(f_k)}{S_n(f_k)} \right) \right\rangle, \\ O_{\text{hSIm}} |k\rangle|0\rangle &= |k\rangle \left| \Im \left(\frac{\tilde{h}(f_k)}{S_n(f_k)} \right) \right\rangle \end{aligned} \tag{46}$$

and

$$\begin{aligned} O_{\text{QRe}} |m\rangle|k\rangle|0\rangle &= |m\rangle|k\rangle \left| \Re \tilde{Q}_m(f_k) \right\rangle, \\ O_{\text{QIm}} |m\rangle|k\rangle|0\rangle &= |m\rangle|k\rangle \left| \Im \tilde{Q}_m(f_k) \right\rangle, \end{aligned} \tag{47}$$

we can combine these along with arithmetic oracles to construct O_{Re} and O_{Im} (note that the remaining factor $2/\Delta t$ is just a known real number independent of m and k).

O_{QRe} and O_{QIm} are in fact implementable. To see this, note that $\tilde{Q}_m(f_k)$ is given as an explicit function of intrinsic parameters and $f_k = k/T$ by theories of GW sources such as CBCs [4–7]. Therefore, if we can relate the index m to intrinsic parameter values by some elementary function, which is in fact possible under simple lattice-like template spacing such as [5], we can write \tilde{Q}_m as an explicit function of m and k and thus construct O_{QRe} and O_{QIm} using arithmetic oracles.

On the other hand, $\tilde{h}(f_k)/S_n(f_k)$ is a factor determined by the experimental data and not represented by an explicit function. We therefore resort to QRAM [53]. This enables us

to access N recorded data x_k labeled by $k \in [N]_0$ and load a specified entry onto a register as

$$|k\rangle|0\rangle \rightarrow |k\rangle|x_k\rangle \quad (48)$$

in superposition in $O(\log N)$ time. It takes $O(M)$ time to register $\{\Re(\frac{\tilde{h}(f_k)}{S_n(f_k)})\}_{k=1,\dots,\frac{M}{2}-1}$ and $\{\Im(\frac{\tilde{h}(f_k)}{S_n(f_k)})\}_{k=1,\dots,\frac{M}{2}-1}$ into a QRAM in advance of running Algorithm 3, but this is expected to be less time-consuming than Algorithm 3 itself, which has $\tilde{O}(\sqrt{N_{\text{temp}}}M)$ query complexity.

3. The mean and variance of detector outputs

Here, we see the validity of Assumption 2 as follows. First, from Eq. (6), for $x_k := 2\Re\tilde{h}(f_k)/\sqrt{S_n(f_k)T}$ and

$y_k := 2\Im\tilde{h}(f_k)/\sqrt{S_n(f_k)T}$ with $k \in [\frac{M}{2}-1]$, we see that $x_1, \dots, x_{\frac{M}{2}-1}, y_1, \dots, y_{\frac{M}{2}-1}$ are independent standard normal variables. We assume that $M \gg 1$ and thus the sample means and the sample variances of $\{x_k\}_k$ and $\{y_k\}_k$ are equal to the population means and the population variances, that is, 0 and 1, respectively. We also assume the usual situation that the signal is much smaller than the noise. More concretely, we assume that, for every $k \in [\frac{M}{2}-1]$, $|u_k| \ll 1$ and $|v_k| \ll 1$ hold, where $u_k := 2\Re\tilde{s}(f_k)/\sqrt{S_n(f_k)T}$ and $v_k := 2\Im\tilde{s}(f_k)/\sqrt{S_n(f_k)T}$, and thus that $|\text{Mean}(\{u_k\}_{k=1,\dots,M/2-1})| \leq 1$, $|\text{Mean}(\{v_k\}_{k=1,\dots,M/2-1})| \leq 1$, $\text{Var}(\{u_k\}_{k=1,\dots,M/2-1}) \leq 1$, and $\text{Var}(\{v_k\}_{k=1,\dots,M/2-1}) \leq 1$ hold. Under these assumptions, we can obtain

$$\begin{aligned} \left| \text{Mean} \left(\left\{ \frac{2\Re\tilde{h}(f_k)}{\sqrt{S_n(f_k)T}} \right\}_{k=1,\dots,M/2-1} \right) \right|^2 &\leq (|\text{Mean}(\{x_k\}_{k=1,\dots,M/2-1})| + |\text{Mean}(\{u_k\}_{k=1,\dots,M/2-1})|)^2 \leq 1, \\ \left| \text{Mean} \left(\left\{ \frac{2\Im\tilde{h}(f_k)}{\sqrt{S_n(f_k)T}} \right\}_{k=1,\dots,M/2-1} \right) \right|^2 &\leq (|\text{Mean}(\{y_k\}_{k=1,\dots,M/2-1})| + |\text{Mean}(\{v_k\}_{k=1,\dots,M/2-1})|)^2 \leq 1, \\ \text{Var} \left(\left\{ \frac{2\Re\tilde{h}(f_k)}{\sqrt{S_n(f_k)T}} \right\}_{k=1,\dots,M/2-1} \right) &\leq (\sqrt{\text{Var}(\{x_k\}_{k=1,\dots,M/2-1})} + \sqrt{\text{Var}(\{u_k\}_{k=1,\dots,M/2-1})})^2 \leq 4, \\ \text{Var} \left(\left\{ \frac{2\Im\tilde{h}(f_k)}{\sqrt{S_n(f_k)T}} \right\}_{k=1,\dots,M/2-1} \right) &\leq (\sqrt{\text{Var}(\{y_k\}_{k=1,\dots,M/2-1})} + \sqrt{\text{Var}(\{v_k\}_{k=1,\dots,M/2-1})})^2 \leq 4. \end{aligned} \quad (49)$$

4. Magnitude of γ

Here, we study the magnitude of γ defined in Eq. (27). Note that the template Q is normalized as $(Q|Q) = 1$, that is

$$\frac{4}{M} \Re \left(\sum_{k=1}^{\frac{M}{2}-1} \frac{|\tilde{Q}_m(f_k)|^2}{S_n(f_k)\Delta t} \right) = 1. \quad (50)$$

Since γ is defined as the square root of the maximum summand in the sum of the left-hand side of Eq. (50), γ can take values in the following range:

$$\frac{1}{\sqrt{2 - \frac{4}{M}}} \leq \gamma \leq \frac{\sqrt{M}}{2}, \quad (51)$$

where the lower bound corresponds to the case in which all summands have the same value and the upper bound corresponds to the case where only one summand contributes. Therefore, we expect that $\gamma = O(1)$ unless $\tilde{Q}_m(f)$ is steeply peaked around some frequency. Let us see whether this holds for the following two types of GWs: those from CBCs and continuous gravitational waves (CGWs).

(A) CBC

If GW interferometers and signals are broadband, we will be closer to the lower bound in Eq. (51) where a significant fraction of the summands have similar values and so γ will be of order $O(1)$.

In Fig. 2, we show an explicit example of this, where we compute the value of γ according to Eq. (27) for the CBC case, modeled using templates Q computed with the IMRPhenomPv2 waveform [56]. The value of γ only depends

on the amplitude evolution of the waveform, which is mostly depend on the component masses, parameterized via the total mass $M = m_1 + m_2$ and the mass ratio $q = m_2/m_1$. We study total masses between $1M_\odot$ and $300M_\odot$, mass ratios between 0.2 and 1, and for this example, we set the spins to 0. For the noise PSD, $S_n(f)$, we use the Advanced LIGO design sensitivity [57]. We assume a sampling rate of 2048 Hz, and low and high frequency cutoffs of 20 Hz and 1024 Hz, respectively.

In Fig. 2, we can observe that γ is of order $O(1)$ in all the parameter space studied. The minimum value of γ is 2.50 at $M = 126M_\odot$, $q = 1.00$, which is the point where the SNR is most homogeneously spread out across frequencies due to the location of the merger (where $|\tilde{Q}(f)|^2 \propto f^{-4/3}$ instead of

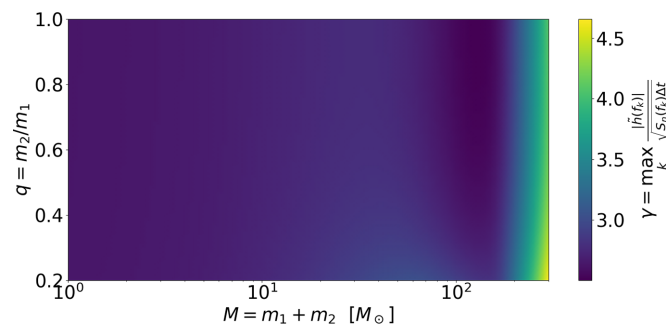


FIG. 2. γ computed with Eq. (27) using the IMRPhenomPv2 waveform [56] with spins set to 0 for the template Q and the Advanced LIGO design sensitivity PSD for $S_n(f)$ [57]. We assume a sampling rate of 2048 Hz, and low- and high-frequency cutoffs of 20 Hz and 1024 Hz, respectively.

$|\tilde{Q}(f)|^2 \propto f^{-7/3}$ as in the inspiral [58]) being just before the most sensitive frequency range of the interferometer. On the other hand, the maximum value of γ is 4.66 at $M = 300M_\odot$, $q = 0.20$. For the very largest masses, we observe that γ tends to increase due to the fact that the higher the mass, the more the template is shifted towards smaller and smaller frequencies, until only the frequencies close to the low frequency cutoff contribute. Nonetheless, γ still takes $O(1)$ values for all masses that can be expected to be seen by the ground-based detectors.

(B) CGW

An all-sky search of the CGWs is known as the most time consuming task in the gravitational wave astronomy (see [59–63] for reviews). CGWs can be modeled by a quasi-monochromatic waveform,

$$Q(t) = A e^{i\Phi(t)}, \quad \Phi(t) = 2\pi f_0 \left(t + \frac{\mathbf{r}(t) \cdot \mathbf{n}}{c} \right), \quad (52)$$

where f_0 is the GW frequency at initial time, $\mathbf{r}(t)$ is the detector's location, and \mathbf{n} is the unit vector pointing toward

the CGW source. Here, we ignore \dot{f} , the change rate of the frequency, just for brevity. First of all, the template must be normalized. The norm of the waveform model (52) is

$$\begin{aligned} N = (Q|Q) &= 2 \int_{-\infty}^{\infty} df \frac{|\tilde{Q}(f)|^2}{S_n(f)} \sim \frac{2}{S_n(f_0)} \int_{-T/2}^{T/2} dt |Q(t)|^2 \\ &= \frac{2A^2 T}{S_n(f_0)}. \end{aligned} \quad (53)$$

Here, the noise PSD is assumed to be constant in the frequency range where the CGW exists. The normalized template is

$$\hat{Q}(t) = \frac{Q(t)}{\sqrt{N}} = \sqrt{\frac{S_n(f_0)}{2T}} e^{i\Phi(t)}. \quad (54)$$

Then the Fourier transform of the normalized template is given by

$$\begin{aligned} \tilde{Q}(f) &= \int_{-T/2}^{T/2} dt \hat{Q}(t) e^{-2\pi i f t} \sim \sqrt{\frac{S_n(f_0)}{2T}} \int_{-T/2}^{T/2} dt \exp \left[2\pi i (f_0 - f) t + 2\pi i f_0 \frac{R_{\text{ES}}}{c} \cos(\Omega_\odot t) \right] \\ &\sim \sqrt{\frac{S_n(f_0)}{2T}} \sum_{n=-\infty}^{\infty} i^n J_n \left(\frac{2\pi f_0 R_{\text{ES}}}{c} \right) \int_{-T/2}^{T/2} dt \exp[2\pi i (f_0 - f) t + i n \Omega_\odot t] \\ &\sim \sqrt{\frac{S_n(f_0)}{2T}} \sum_{n=-\infty}^{\infty} i^n J_n \left(\frac{2\pi f_0 R_{\text{ES}}}{c} \right) \frac{\sin[\pi(f_0 - f)T + n\Omega_\odot T/2]}{\pi(f_0 - f) + n\Omega_\odot/2} \sim \sqrt{\frac{S_n(f_0)}{2T}} \sqrt{\frac{2}{\pi}} \cdot \frac{c}{2\pi f_0 R_{\text{ES}}} \cdot T. \end{aligned} \quad (55)$$

Here we approximated the detector's motion by the orbital motion of the earth: $\mathbf{r}(t) = R_{\text{ES}} \cos(\Omega_\odot t)$, where R_{ES} is the distance between the sun and the earth, and Ω_\odot is the angular velocity of the earth's orbital motion. We used the Jacobi-Anger expansion

$$e^{iz \cos \theta} = \sum_{n=-\infty}^{\infty} i^n J_n(z) e^{in\theta}, \quad (56)$$

and the asymptotic form of the Bessel function

$$J_n(z) \rightarrow \sqrt{\frac{2}{\pi z}} \cos \left(z - \frac{2n+1}{4} \pi \right) \quad \text{for } |z| \rightarrow \infty. \quad (57)$$

In the last line, we assume $f = f_0$ and the $n = 0$ term dominates the other terms. Therefore, the γ factor is estimated as

$$\gamma \sim \frac{|\tilde{Q}(f)|}{\sqrt{S_n(f_0) \Delta t}} \sim \sqrt{\frac{T}{\Delta t}} \cdot \frac{c}{2\pi f_0 R_{\text{ES}}}. \quad (58)$$

Assuming the typical values, $\Delta t \sim 10^{-3}$ sec, $T \sim 10^7$ sec, and $f_0 = 10^2$ Hz, we get $\gamma \sim O(10^2)$.

Thus, γ for CGWs typically becomes larger than that in the CBC case, but our quantum algorithm is still expected to be faster than the classical method in terms of query complexity because of the large template number in a CGW search, as we see in the followings. To estimate the grid points taken in the template parameter space, we now focus on the grid

number on the sky, that is, the unit sphere of \mathbf{n} . Considering the detector's motion with respect to the CGW source, the sufficiently fine angular resolution is roughly estimated by

$$\Delta\theta \sim \frac{\lambda}{2R_{\text{ES}}} \sim 10^{-5} [\text{rad}], \quad (59)$$

where λ is the wavelength of CGWs. Thus, we need $N_{\text{grid}} \sim 4\pi/(\Delta\theta)^2 \sim 10^{11}$ grid points to cover the entire sky. The ratio of the query complexity of our algorithm to the classical one is $O(\gamma/\sqrt{N_{\text{grid}}})$. For $\gamma = 10^2$ and $N_{\text{grid}} = 10^{11}$, this factor is much smaller than 1, which means that our algorithm still provides the quantum speedup. Note that, the number of grid points for the template search increases if we consider not only \mathbf{n} but also f_0 and \dot{f} , in which case the advantage of our algorithm becomes more prominent.

Here, we assume the coherent approach in which we simultaneously analyze the entire strain data. In reality, because of limitation in the computational resources, the standard today's commonly-used approach is the semicoherent one, in which we divide the strain data into many short segments, carry out matched filtering for each segment, and combine them. Because most of the semicoherent methods are based on the matched filtering approach, our algorithm would be applicable for the semicoherent methods. Naively speaking, semicoherent search has fewer templates, so we may have less gain in quantum speedup, but it may be compensated by a smaller

value of γ . However, because of the more complicated procedure in the semicoherent methods, it is not straightforward to estimate the total gain, so we postpone the discussion on this point to future works.

IV. SUMMARY

In this paper, we proposed a quantum algorithm for GW matched filtering based on Ref. [3], which has investigated the application of Grover's search algorithm. Our method, described in details in Sec. III C, uses QMCI for the SNR calculation of Eq. (11) instead of FFT used in the classical method and in Ref. [3], and searches high SNR templates by QAA, running QMCI as a subroutine. To deal with the erroneous nature of QMCI, we propose to set two thresholds ρ_{hard} and ρ_{soft} such that the proposed algorithm returns "there is a signal" for events with SNR $\rho \geq \rho_{\text{hard}}$ with high probability and "there is no signal" for events with SNR $\rho < \rho_{\text{soft}}$ with certainty.

Our main results are summarized in Eqs. (40), (41), and (42). By neglecting the logarithmic dependencies, we can see that the proposed algorithm has $\tilde{O}(M/\sqrt{r(\rho_{\text{th}})})$ and $\tilde{O}(M\sqrt{N_{\text{temp}}})$ query complexity for $\tilde{r}(\rho_{\text{hard}}) > 0$ and $\tilde{r}(\rho_{\text{hard}}) = 0$, respectively. This still indicates a quadratic speedup with respect to the template number N_{temp} and the same order of complexity with respect to the number of time-series data points M compared to the algorithm of Ref. [3], which is summarized in Sec. III B. We note that the choice of ρ_{hard} and ρ_{soft} , the accuracy of QMCI in other words, mildly affects the complexity, but according to the discussion in Sec. III D 1, we take $\rho_{\text{hard}} - \rho_{\text{soft}} = 1$, and thus not changing the factor.

The advantage of this algorithm is that it requires only qubit number that logarithmically scales on M as described in Eq. (39), contrary to the FFT-based method that requires $O(M)$ qubits. Therefore, this algorithm is expected to be beneficial in the situation that quantum computers have a limitation on qubit number, which is likely to occur due to the large overhead for quantum error correction. A possible drawback of the proposed algorithm compared to the algorithm in [3] is that, to load the detector output data onto a register in superposition in QMCI, it uses QRAM, whose experimental realization is challenging [64].

In any case, we believe that proposing multiple ways of applying quantum methods that have different pros and cons is highly meaningful in taking advantage of quantum computing in future GW experiments, given today's uncertainty on what the future quantum computers will be. Discussions for applying quantum computing in experimental physics have just started. We anticipate that more proposals will follow not only for GW data analysis but also for other heavy data analyses in various physical and astronomical experiments.

ACKNOWLEDGMENTS

G.M., S.K., and S.N. acknowledge support from the research Project No. PGC2018-094773-B-C32, and the Spanish Research Agency (Agencia Estatal de Investigación) through the Grant IFT Centro de Excelencia Severo Ochoa No CEX2020-001007-S, funded by

MCIN/AEI/10.13039/501100011033. S.K. is supported by the Spanish Atracción de Talento Contract No. 2019-T1/TIC-13177 granted by Comunidad de Madrid, the I+D Grant No. PID2020-118159GA-C42 of the Spanish Ministry of Science and Innovation and the i-LINK 2021 Grant No. LINKA20416 of CSIC. T.Y. and S.K. are supported by Japan Society for the Promotion of Science (JSPS) KAKENHI Grants No. JP20H01899 and No. JP20H05853. K.M. is supported by MEXT Quantum Leap Flagship Program (MEXT Q-LEAP) Grant No. JPMXS0120319794 and JSPS KAKENHI Grant No. JP22K11924.

APPENDIX A: PROOF OF THEOREM 5

The quantum algorithm for estimating the mean of variables with bounded variance is given as Algorithm 3 in [15]. In order to use it as a subroutine in QAA, we now want to present it as a unitary transformation. That is, we aim to remove measurements in the original algorithm in [15]. Then, the modified algorithm outputs a quantum state in which the computational basis states corresponding to approximations of the mean have squared amplitudes summing up to almost 1.

We first present the following theorem on the mean estimation method for a bounded variable. The method and the theorem are almost the same as Algorithm 1 and Theorem 2.3 in [15] in a specific case. But, to be self-contained, we now present them with proof. Note that, although Algorithm 1 in [15] is a procedure containing measurements, the following is a unitary transformation with no measurement to generate some quantum state.

Theorem 7. Let $N \in \mathbb{N}$ and \mathcal{X} be a set of N real numbers $X_0, \dots, X_{N-1} \in [0, 1]$. Suppose that we are given an oracle $O_{\mathcal{X}}$ that acts as Eq. (22). Then, for any integer t larger than 2, there is an oracle $O_{\mathcal{X},t}^{\text{mean}}$ that acts as Eq. (23), where some ancillary qubits are undisplayed. Here, \mathcal{Y} is a finite set of real numbers that includes a subset $\tilde{\mathcal{Y}}$ consisting of elements $\tilde{\mu}$ satisfying

$$|\tilde{\mu} - \mu| \leq C \left(\frac{\sqrt{\mu}}{t} + \frac{1}{t^2} \right), \quad (\text{A1})$$

with $\mu = \frac{1}{N} \sum_{i=0}^{N-1} X_i$ and a universal real constant C , and $\{\alpha_y\}_{y \in \mathcal{Y}}$ are complex numbers satisfying $\sum_{y \in \tilde{\mathcal{Y}}} |\alpha_y|^2 \geq 8/\pi^2$. In $O_{\mathcal{X}}$, $O(t)$ queries to $O_{\mathcal{X}}$ are made and $O(\log N + \log t)$ ancillary qubits are used.

Proof. Combining O_N^{EqPr} and $O_{\mathcal{X}}$, we can construct an oracle $O'_{\mathcal{X}}$ on a system with $O(\log N)$ qubits in total such that

$$O'_{\mathcal{X}}|0\rangle|0\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle|X_i\rangle, \quad (\text{A2})$$

and, combining this with some arithmetic oracles, we can construct an oracle $O''_{\mathcal{X}}$ that acts as

$$O''_{\mathcal{X}}|0\rangle|0\rangle|0\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle|X_i\rangle(\sqrt{X_i}|1\rangle + \sqrt{1-X_i}|0\rangle), \quad (\text{A3})$$

Algorithm 4. QMCI algorithm with measurements.

Input: oracle $O_{\mathcal{X}}$ in Eq. (22), an upper bound σ of $\text{Var}(\mathcal{X})$, accuracy $\epsilon \in (0, 4\sigma)$, and $\delta \in (0, 1)$

- 1: Set $L := \lceil \log_2(\frac{32\sigma}{\epsilon}) \rceil$, $J := 12\lceil \log \delta^{-1} \rceil + 1$, $K := 12\lceil \log(\frac{64(L+1)}{5}) \rceil + 1$, $t_0 := \lceil \frac{32\sigma D \sqrt{\log_2(\frac{32\sigma}{\epsilon})}}{\epsilon} \rceil$, where D is a universal constant given in [15].
- 2: **for** $j = 1, \dots, J$ **do**
- 3: Randomly choose an integer i_j from $[N]_0$ and generate the state $O_{\mathcal{X}}|i_j\rangle|0\rangle = |i_j\rangle|X_{i_j}\rangle$. Measure the second register in the computational basis and let the outcome divided by σ be \tilde{m}_j .
- 4: **for** $l = 0, 1, \dots, L$ **do**
- 5: Let $\tilde{\mathcal{X}}_{j,l}^+ := \{\tilde{X}_{l,1}^+(\tilde{m}_j), \dots, \tilde{X}_{l,N}^+(\tilde{m}_j)\}$ and $\tilde{\mathcal{X}}_{j,l}^- := \{\tilde{X}_{l,1}^-(\tilde{m}_j), \dots, \tilde{X}_{l,N}^-(\tilde{m}_j)\}$, where, for $i \in [N]_0$ and $m \in \mathbb{R}$, $\tilde{X}_{l,i}^\pm(m)$ is given as

$$\begin{aligned} \tilde{X}_{0,i}^+(m) &:= \begin{cases} \frac{1}{4}(\frac{X_i}{\sigma} - \tilde{m}); & \text{if } 0 \leq \frac{1}{4}(\frac{X_i}{\sigma} - \tilde{m}) < 1 \\ 0; & \text{otherwise} \end{cases}, \\ \tilde{X}_{0,i}^-(m) &:= \begin{cases} -\frac{1}{4}(\frac{X_i}{\sigma} - \tilde{m}); & \text{if } -1 < \frac{1}{4}(\frac{X_i}{\sigma} - \tilde{m}) < 0 \\ 0; & \text{otherwise} \end{cases} \end{aligned} \quad (\text{A7})$$

when $l = 0$, and

$$\begin{aligned} \tilde{X}_{l,i}^+(m) &:= \begin{cases} \frac{1}{4 \cdot 2^l}(\frac{X_i}{\sigma} - \tilde{m}); & \text{if } 2^{l-1} \leq \frac{1}{4}(\frac{X_i}{\sigma} - \tilde{m}) < 2^l \\ 0; & \text{otherwise} \end{cases}, \\ \tilde{X}_{l,i}^-(m) &:= \begin{cases} -\frac{1}{4 \cdot 2^l}(\frac{X_i}{\sigma} - \tilde{m}); & \text{if } -2^l < \frac{1}{4}(\frac{X_i}{\sigma} - \tilde{m}) \leq -2^{l-1} \\ 0; & \text{otherwise} \end{cases} \end{aligned} \quad (\text{A8})$$

when $l \geq 1$.

- 6: Construct an oracle $O_{\tilde{\mathcal{X}}_{j,l}^+}$ (resp. $O_{\tilde{\mathcal{X}}_{j,l}^-}$) such that $O_{\tilde{\mathcal{X}}_{j,l}^+}|0\rangle|0\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle|\tilde{X}_{l,i}^+(m_j)\rangle$ [resp. $O_{\tilde{\mathcal{X}}_{j,l}^-}|0\rangle|0\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle|\tilde{X}_{l,i}^-(m_j)\rangle$] by combining O_N^{EqPr} , $O_{\mathcal{X}}$ and some arithmetic oracles.
- 7: Using $O_{\tilde{\mathcal{X}}_{j,l}^+}$, construct an oracle $O_{\tilde{\mathcal{X}}_{j,l}^+, t_0}^{\text{mean}}$ that acts like Eq.(23), that is, $O_{\tilde{\mathcal{X}}_{j,l}^+, t_0}^{\text{mean}}|0\rangle = \sum_{y \in \mathcal{Y}_{j,l}^+} \alpha_y |y\rangle$, where a real number set $\mathcal{Y}_{j,l}^+$ includes a subset $\tilde{\mathcal{Y}}_{j,l}^+$ such that, for every $\tilde{\mu} \in \tilde{\mathcal{Y}}_{j,l}^+$, $|\tilde{\mu} - \mu_{j,l}^+| \leq C(\frac{\sqrt{\mu_{j,l}^+}}{t_0} + \frac{1}{t_0})$ holds with $\mu_{j,l}^+ := \frac{1}{N} \sum_{i=0}^{N-1} \tilde{X}_{l,i}^+$ and that $\sum_{\tilde{\mu} \in \tilde{\mathcal{Y}}_{j,l}^+} |\alpha_{\tilde{\mu}}|^2 \geq 8/\pi^2$. Similarly, using $O_{\tilde{\mathcal{X}}_{j,l}^-}$, construct an oracle $O_{\tilde{\mathcal{X}}_{j,l}^-, t_0}^{\text{mean}}$ that acts as $O_{\tilde{\mathcal{X}}_{j,l}^-, t_0}^{\text{mean}}|0\rangle = \sum_{y \in \mathcal{Y}_{j,l}^-} \alpha_y |y\rangle$, where a real number set $\mathcal{Y}_{j,l}^-$ includes a subset $\tilde{\mathcal{Y}}_{j,l}^-$ such that, for every $\tilde{\mu} \in \tilde{\mathcal{Y}}_{j,l}^-$, $|\tilde{\mu} - \mu_{j,l}^-| \leq C(\frac{\sqrt{\mu_{j,l}^-}}{t_0} + \frac{1}{t_0})$ holds with $\mu_{j,l}^- := \frac{1}{N} \sum_{i=0}^{N-1} \tilde{X}_{l,i}^-$ and that $\sum_{\tilde{\mu} \in \tilde{\mathcal{Y}}_{j,l}^-} |\alpha_{\tilde{\mu}}|^2 \geq 8/\pi^2$.
- 8: Generate K copies of the quantum state $O_{\tilde{\mathcal{X}}_{j,l}^+, t_0}^{\text{mean}}|0\rangle$ (resp. $O_{\tilde{\mathcal{X}}_{j,l}^-, t_0}^{\text{mean}}|0\rangle$) and measure them in the computational basis. Let the median of the measurement outcomes be $\tilde{\mu}_{j,l}^+$ (resp. $\tilde{\mu}_{j,l}^-$).
- 9: **end for**
- 10: Set $\tilde{\mu}_j := \sigma(\tilde{m}_j + 4 \sum_{l=0}^L 2^l (\tilde{\mu}_{j,l}^+ - \tilde{\mu}_{j,l}^-))$.
- 11: **end for**
- 12: Output the median of $\tilde{\mu}_1, \dots, \tilde{\mu}_J$.

where the last ket corresponds to an ancillary qubit. Note that we can write the right-hand side as

$$\sqrt{\mu}|\Phi_1\rangle|1\rangle + \sqrt{1-\mu}|\Phi_0\rangle|0\rangle, \quad (\text{A4})$$

where $|\Phi_1\rangle$ and $|\Phi_0\rangle$ are some quantum states on the first two registers. Then, as stated in Theorem 3, using $O_{\mathcal{X}}'$ $O(t)$ times, we can construct $O_{\mathcal{X},t}^{\text{mean}}$ that acts as

$$O_{\mathcal{X},t}^{\text{mean}}|0\rangle = \sum_{y \in \mathcal{Y}} \alpha_y |y\rangle, \quad (\text{A5})$$

where a real number set \mathcal{Y} includes a subset $\tilde{\mathcal{Y}}$ such that, for every $\tilde{\mu} \in \tilde{\mathcal{Y}}$,

$$|\tilde{\mu} - \mu| \leq \frac{2\pi\sqrt{\mu(1-\mu)}}{t} + \frac{\pi^2}{t^2} \leq \pi^2 \left(\frac{\sqrt{\mu}}{t} + \frac{1}{t^2} \right) \quad (\text{A6})$$

holds and that $\sum_{\tilde{\mu} \in \tilde{\mathcal{Y}}} |\alpha_{\tilde{\mu}}|^2 \geq 8/\pi^2$. Since $O_{\mathcal{X}}'$ contains one query to $O_{\mathcal{X}}$ and uses $O(\log N)$ qubits, the statements on the

query complexity and the qubit number immediately follows from Theorem 3. \blacksquare

We then prove Theorem 5.

Proof of Theorem 5. Our Theorem 5 is almost the same as Theorem 2.5 in [15] in a specific case and we now just aim to modify Algorithm 3 in [15] so that all intermediate measurements are removed and that the output is a quantum state, see Eq. (23). Let us start by presenting the method with measurements as Algorithm 4, which outputs an ϵ approximation of μ , according to Theorem 2.5 in [15]. Note that, although the procedure looks different from that in Algorithm 3 in [15], it is actually almost the same and the difference arises just because we explicitly write the steps that are originally shown separately as Algorithm 2 in [15], in lines 7 and 8. There are only two differences, which enhance the lower bound of the success probability from $\frac{2}{3}$, the original value in [15], to $1 - \delta$. First, K , the number of repeated state generations and measurements in line 8, is different from that in Algorithm

2 in [15]. The current setting of K makes the probability that, for each $(j, l) \in [J] \times [L + 1]_0$, $\tilde{\mu}_{j,l}^+$ (resp. $\tilde{\mu}_{j,l}^-$) becomes an estimate of $\mu_{j,l}^+$ (resp. $\mu_{j,l}^-$) with desired accuracy larger than $1 - \frac{5}{64(L+1)}$ (see Theorem 4). Thus, the probability that $2(L + 1)$ estimations of $\mu_{j,0}^+, \dots, \mu_{j,L}^+, \mu_{j,0}^-, \dots, \mu_{j,L}^-$ simultaneously succeed is larger than $(1 - \frac{5}{64(L+1)})^{2(L+1)} \geq \frac{27}{32}$, and therefore, the probability that $\tilde{\mu}_j$ is close to μ is larger than $\frac{8}{9} \times \frac{27}{32} = \frac{3}{4}$ ($\frac{8}{9}$ is a lower bound on the probability for $|\sigma \tilde{m}_j - \mu| \leq 3\sigma$; see [15]). Second, Algorithm 4 in this paper has the loop on $j = 1, \dots, J$, which means that Algorithm 3 in [15] itself is repeated J times and that the median of the outputs is taken. This makes the lower bound of the success probability of the μ estimation from $\frac{3}{4}$ to $1 - \delta$.

Now, let us present the implementation of this algorithm without intermediate measurements. For a preparation, we randomly choose J integers from $[N]_0$ and let them i_1, \dots, i_J . Then, on a four-register system initialized as $|i_j\rangle|0\rangle|0\rangle|0\rangle$, we perform $O_{\mathcal{X}}$ and a division to yield $|i_j\rangle|\tilde{m}_j\rangle|0\rangle|0\rangle$, where $\tilde{m}_j := X_{i_j}/\sigma$. Furthermore, combining O_N^{EqPr} , $O_{\mathcal{X}}$ and some arithmetic oracles, we construct an oracle $O_{\tilde{\mathcal{X}}_{j,l}^+}$ that acts on $|i_j\rangle|\tilde{m}_j\rangle|0\rangle|0\rangle$ as

$$O_{\tilde{\mathcal{X}}_{j,l}^+} |i_j\rangle|\tilde{m}_j\rangle|0\rangle|0\rangle = |i_j\rangle|\tilde{m}_j\rangle \left(\frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle|\tilde{X}_{l,i}^+(\tilde{m}_j)\rangle \right). \quad (\text{A9})$$

Similarly, we obtain an oracle $O_{\tilde{\mathcal{X}}_{j,l}^-}$ that acts as

$$O_{\tilde{\mathcal{X}}_{j,l}^-} |i_j\rangle|\tilde{m}_j\rangle|0\rangle|0\rangle = |i_j\rangle|\tilde{m}_j\rangle \left(\frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle|\tilde{X}_{l,i}^-(\tilde{m}_j)\rangle \right). \quad (\text{A10})$$

Using $O_{\tilde{\mathcal{X}}_{j,l}^+}$ and $O_{\tilde{\mathcal{X}}_{j,l}^-}$ $O(t_0)$ times, we can construct oracles $\tilde{O}_{\tilde{\mathcal{X}}_{j,l}^+,t_0}^{\text{mean}}$ and $\tilde{O}_{\tilde{\mathcal{X}}_{j,l}^-,t_0}^{\text{mean}}$, which resemble $O_{\tilde{\mathcal{X}}_{j,l}^+,t_0}^{\text{mean}}$ and $O_{\tilde{\mathcal{X}}_{j,l}^-,t_0}^{\text{mean}}$ in Algorithm 4, respectively, but act as

$$\tilde{O}_{\tilde{\mathcal{X}}_{j,l}^+,t_0}^{\text{mean}} |i_j\rangle|\tilde{m}_j\rangle|0\rangle = |i_j\rangle|\tilde{m}_j\rangle \left(\sum_{y \in \mathcal{Y}_{j,l}^+} \alpha_y |y\rangle \right), \quad (\text{A11})$$

$$\tilde{O}_{\tilde{\mathcal{X}}_{j,l}^-,t_0}^{\text{mean}} |i_j\rangle|\tilde{m}_j\rangle|0\rangle = |i_j\rangle|\tilde{m}_j\rangle \left(\sum_{y \in \mathcal{Y}_{j,l}^-} \alpha_y |y\rangle \right),$$

where $\mathcal{Y}_{j,l}^\pm$ and α_y are described in Algorithm 4 and some registers are not displayed. Then, on an appropriate number of registers, some of which are initialized to $|i_1\rangle, \dots, |i_J\rangle$, we use $O_{\mathcal{X}}$, $\tilde{O}_{\tilde{\mathcal{X}}_{j,l}^+,t_0}^{\text{mean}}$ and $\tilde{O}_{\tilde{\mathcal{X}}_{j,l}^-,t_0}^{\text{mean}}$ to generate the following quantum state:

$$\begin{aligned} & \bigotimes_{j=1}^J |i_j\rangle|\tilde{m}_j\rangle \otimes \left[\bigotimes_{l=0}^L \left(\sum_{y_{j,l,1}^+ \in \mathcal{Y}_{j,l}^+} \alpha_{y_{j,l,1}^+} |y_{j,l,1}^+\rangle \right) \otimes \dots \otimes \left(\sum_{y_{j,l,K}^+ \in \mathcal{Y}_{j,l}^+} \alpha_{y_{j,l,K}^+} |y_{j,l,K}^+\rangle \right) \right. \\ & \left. \otimes \left(\sum_{y_{j,l,1}^- \in \mathcal{Y}_{j,l}^-} \alpha_{y_{j,l,1}^-} |y_{j,l,1}^-\rangle \right) \otimes \dots \otimes \left(\sum_{y_{j,l,K}^- \in \mathcal{Y}_{j,l}^-} \alpha_{y_{j,l,K}^-} |y_{j,l,K}^-\rangle \right) \right] \\ & = \bigotimes_{j=1}^J |i_j\rangle|\tilde{m}_j\rangle \otimes \left[\bigotimes_{l=0}^L \sum_{\substack{y_{j,l,1}^+ \in \mathcal{Y}_{j,l}^+, \dots, y_{j,l,K}^+ \in \mathcal{Y}_{j,l}^+ \\ y_{j,l,1}^- \in \mathcal{Y}_{j,l}^-, \dots, y_{j,l,K}^- \in \mathcal{Y}_{j,l}^-}} \left(\prod_{k=1}^K \alpha_{y_{j,l,k}^+} \alpha_{y_{j,l,k}^-} \right) |y_{j,l,1}^+\rangle \dots |y_{j,l,K}^+\rangle |y_{j,l,1}^-\rangle \dots |y_{j,l,K}^-\rangle \right]. \quad (\text{A12}) \end{aligned}$$

Further, adding some registers and performing O_K^{med} , we obtain

$$\bigotimes_{j=1}^J |i_j\rangle|\tilde{m}_j\rangle \otimes \left[\bigotimes_{l=0}^L \sum_{\substack{y_{j,l,1}^+ \in \mathcal{Y}_{j,l}^+, \dots, y_{j,l,K}^+ \in \mathcal{Y}_{j,l}^+ \\ y_{j,l,1}^- \in \mathcal{Y}_{j,l}^-, \dots, y_{j,l,K}^- \in \mathcal{Y}_{j,l}^-}} \left(\prod_{k=1}^K \alpha_{y_{j,l,k}^+} \alpha_{y_{j,l,k}^-} \right) |y_{j,l,1}^+\rangle \dots |y_{j,l,K}^+\rangle |y_{j,l,1}^-\rangle \dots |y_{j,l,K}^-\rangle |\tilde{\mu}_{j,l}^+\rangle |\tilde{\mu}_{j,l}^-\rangle \right], \quad (\text{A13})$$

where $\tilde{\mu}_{j,l}^+ = \text{med}(y_{j,l,1}^+, \dots, y_{j,l,K}^+)$ and $\tilde{\mu}_{j,l}^- = \text{med}(y_{j,l,1}^-, \dots, y_{j,l,K}^-)$. Moreover, adding further registers and using arithmetic oracles, we obtain

$$\bigotimes_{j=1}^J |i_j\rangle|\tilde{m}_j\rangle \otimes \left\{ \left[\bigotimes_{l=0}^L \sum_{\substack{y_{j,l,1}^+ \in \mathcal{Y}_{j,l}^+, \dots, y_{j,l,K}^+ \in \mathcal{Y}_{j,l}^+ \\ y_{j,l,1}^- \in \mathcal{Y}_{j,l}^-, \dots, y_{j,l,K}^- \in \mathcal{Y}_{j,l}^-}} \left(\prod_{k=1}^K \alpha_{y_{j,l,k}^+} \alpha_{y_{j,l,k}^-} \right) |y_{j,l,1}^+\rangle \dots |y_{j,l,K}^+\rangle |y_{j,l,1}^-\rangle \dots |y_{j,l,K}^-\rangle |\tilde{\mu}_{j,l}^+\rangle |\tilde{\mu}_{j,l}^-\rangle \right] \otimes |\tilde{\mu}_j\rangle \right\}, \quad (\text{A14})$$

where $\tilde{\mu}_j := \sigma(\tilde{m}_j + 4 \sum_{l=0}^L 2^l (\tilde{\mu}_{j,l}^+ - \tilde{\mu}_{j,l}^-))$. Finally, performing O_J^{med} yields the state

$$\left\{ \bigotimes_{j=1}^J |i_j\rangle |\tilde{m}_j\rangle \otimes \left[\left(\bigotimes_{l=0}^L \sum_{\substack{y_{j,l,1}^+ \in \mathcal{Y}_{j,l,1}^+, \dots, y_{j,l,K}^+ \in \mathcal{Y}_{j,l,K}^+ \\ y_{j,l,1}^- \in \mathcal{Y}_{j,l,1}^-, \dots, y_{j,l,K}^- \in \mathcal{Y}_{j,l,K}^-}} \left(\prod_{k=1}^K \alpha_{y_{j,l,k}^+} \alpha_{y_{j,l,k}^-} \right) |y_{j,l,1}^+\rangle \cdots |y_{j,l,K}^+\rangle |y_{j,l,1}^-\rangle \cdots |y_{j,l,K}^-\rangle |\tilde{\mu}_{j,l}^+\rangle |\tilde{\mu}_{j,l}^-\rangle \right) \otimes |\tilde{\mu}_j\rangle \right] \right\} \otimes |\tilde{\mu}\rangle, \quad (\text{A15})$$

where $\tilde{\mu} = \text{med}(\tilde{\mu}_1, \dots, \tilde{\mu}_J)$. As Algorithm 4, if we measure the last register in this final state, we obtain an ϵ -approximation of μ with probability at least $1 - \delta$. This means that the final state, Eq. (A15), can be written as Eq. (23), with some registers undisplayed. We therefore regard the above unitary transformation to generate Eq. (A15) as $O_{\mathcal{X}, \epsilon, \delta, \sigma}^{\text{mean}}$.

Lastly, let us consider the statements on the query complexity and the qubit number. Note that we generate the state of Eq. (A12) by $O(J)$ uses of $O_{\mathcal{X}}$ and

$$O(JKL) = O\left[\log\left(\frac{\sigma}{\epsilon}\right) \log\log\left(\frac{\sigma}{\epsilon}\right) \log\delta^{-1}\right] \quad (\text{A16})$$

uses of $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^+}^{\text{mean}}$ and $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^-}^{\text{mean}}$, along with initialization of some registers to $|i_1\rangle, \dots, |i_J\rangle$, and the transformation from Eq. (A12) to Eq. (A15) is done by only arithmetic oracles. Also note that, in $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^+}^{\text{mean}}$ and $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^-}^{\text{mean}}$, $O_{\tilde{\mathcal{X}}_{j,l}^+}$, and $O_{\tilde{\mathcal{X}}_{j,l}^-}$ are called $O(t_0) = O[\frac{\sigma}{\epsilon} \log^{1/2}(\frac{\sigma}{\epsilon})]$ times. The number of calls to $O_{\mathcal{X}}$ in $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^+}^{\text{mean}}$ and $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^-}^{\text{mean}}$ is also of the same order, since each of $O_{\tilde{\mathcal{X}}_{j,l}^+}$ and $O_{\tilde{\mathcal{X}}_{j,l}^-}$ contains one call to $O_{\mathcal{X}}$. Combining these observations, we see that the number of uses of $O_{\mathcal{X}}$ in generating Eq. (A15) is given by Eq. (24). When it comes to qubit number, we note that, as stated in Theorem 7, $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^+}^{\text{mean}}$ and $\tilde{O}_{\tilde{\mathcal{X}}_{j,l,t_0}^-}^{\text{mean}}$ use $O(\log N + \log t_0) = O(\log N + \log(\frac{\sigma}{\epsilon}))$ qubits, and thus we use $O((\log N + \log(\frac{\sigma}{\epsilon})) \times JKL)$ qubits, which is of order as in Eq. (25), in preparing Eq. (A12). Also note that added registers in transformation from Eq. (A12) to Eq. (A15) is $O(JL)$. From these observations, the total number of qubits used in generating Eq. (A15) is given by Eq. (25). ■

APPENDIX B: REMAINING PART OF THE PROOF OF THEOREM 6

The remaining part of the proof.

Query complexity and qubit number

We consider the following cases separately.

(i) $r(\rho_{\text{hard}}) > 0$

In this case, there exists $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$ such that $\rho_{m,j} \geq \rho_{\text{hard}}$. For such (m, j) ,

$$|y - \rho_{m,j}| \leq \epsilon \Rightarrow y \geq \rho_{\text{mid}} \quad (\text{B1})$$

holds for any $y \in \mathbb{R}$ (recall that we are now setting $\epsilon = \frac{\rho_{\text{hard}} - \rho_{\text{soft}}}{2}$), and thus

$$\sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2 \geq \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ |y - \rho_{m,j}| \leq \epsilon}} |\alpha_{\rho,y}|^2 \geq 1 - \delta' \geq \frac{1}{2} \quad (\text{B2})$$

holds. Using this, p_1 , the probability that we obtain 1 when we measure the last qubit in the final state in Eq. (45), is evaluated as

$$\begin{aligned} p_1 &= \frac{1}{N_{\text{temp}} M} \sum_{m=0}^{N_{\text{temp}}-1} \sum_{j=0}^{M-1} \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2 \\ &\geq \frac{1}{N_{\text{temp}} M} \sum_{\substack{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0 \\ \rho_{m,j} \geq \rho_{\text{hard}}}} \frac{1}{2} \\ &= \frac{\tilde{r}(\rho_{\text{hard}})}{2} \geq \frac{1}{2N_{\text{temp}} M}. \end{aligned} \quad (\text{B3})$$

Therefore, by $\text{QAA}(O_{\text{AE}}, \frac{1}{2N_{\text{temp}} M}, \frac{\delta}{2})$, we get the state

$$|\psi\rangle := \frac{1}{\sqrt{p_1}} \sum_{m=0}^{N_{\text{temp}}-1} \sum_{j=0}^{M-1} |m\rangle |j\rangle \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} \alpha_{\rho,y} |y\rangle, \quad (\text{B4})$$

with probability at least $1 - \frac{\delta}{2}$.

On the other hand, note that, for $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$ such that $\rho_{m,j} < \rho_{\text{soft}}$,

$$y \geq \rho_{\text{mid}} \Rightarrow |y - \rho_{m,j}| > \epsilon \Rightarrow y \notin \tilde{\mathcal{Y}}_{m,j} \quad (\text{B5})$$

holds for any $y \in \mathbb{R}$, and thus we have

$$\sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2 = \sum_{\substack{y \in \mathcal{Y}_{m,j} \setminus \tilde{\mathcal{Y}}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2 \leq \sum_{y \in \mathcal{Y}_{m,j} \setminus \tilde{\mathcal{Y}}_{m,j}} |\alpha_{\rho,y}|^2 < \delta'. \quad (\text{B6})$$

This means that $p_{\geq \rho_{\text{soft}}}$ the probability that we obtain (m, j) such that $\rho_{m,j} \geq \rho_{\text{soft}}$ when we measure the first two registers in $|\psi\rangle$ is evaluated as

$$\begin{aligned} p_{\geq \rho_{\text{soft}}} &= \frac{1}{p_1} \sum_{\substack{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0 \\ \rho_{m,j} \geq \rho_{\text{soft}}}} \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} \frac{1}{N_{\text{temp}} M} |\alpha_{\rho,y}|^2 \\ &= \frac{\sum_{\substack{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0 \\ \rho_{m,j} \geq \rho_{\text{soft}}}} \sum_{y \in \mathcal{Y}_{m,j}} |\alpha_{\rho,y}|^2}{\sum_{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0} \sum_{y \in \mathcal{Y}_{m,j}} |\alpha_{\rho,y}|^2} \\ &= 1 - \frac{\sum_{\substack{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0 \\ \rho_{m,j} < \rho_{\text{soft}}}} \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2}{\sum_{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0} \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2} \\ &\geq 1 - 2N_{\text{temp}} M \delta' = 1 - \frac{\delta}{2}, \end{aligned} \quad (\text{B7})$$

where, at the inequality, we use

$$\sum_{\substack{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0 \\ \rho_{m,j} < \rho_{\text{soft}}}} \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2 \leq \sum_{\substack{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0 \\ \rho_{m,j} < \rho_{\text{soft}}}} \delta' \leq \sum_{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0} \delta' = N_{\text{temp}} M \delta' \tag{B8}$$

and

$$\sum_{(m,j) \in [N_{\text{temp}}]_0 \times [M]_0} \sum_{\substack{y \in \mathcal{Y}_{m,j} \\ y \geq \rho_{\text{mid}}}} |\alpha_{\rho,y}|^2 = N_{\text{temp}} M p_1 \geq \frac{1}{2}. \tag{B9}$$

In summary, by the algorithm, we get (m, j) such that $\rho_{m,j} \geq \rho_{\text{soft}}$ with probability at least $(1 - \frac{\delta}{2})^2 \geq 1 - \delta$.

The query complexity is evaluated as follows. Until we get an output, $\text{QAA}(O_{\text{AE}}, \frac{1}{2N_{\text{temp}}M}, \frac{\delta}{2})$ calls O_{AE}

$$O\left(\frac{\log \delta^{-1}}{\sqrt{p_1}}\right) = O\left(\frac{\log \delta^{-1}}{\sqrt{\tilde{r}(\rho_{\text{hard}})}}\right) \tag{B10}$$

times. As stated in Theorem 5, O_{AE} makes

$$O\left[\frac{\sigma}{\epsilon} \log^{3/2}\left(\frac{\sigma}{\epsilon}\right) \log \log\left(\frac{\sigma}{\epsilon}\right) \log\left(\frac{1}{\delta'}\right)\right] = O\left[\frac{\sqrt{M}\gamma}{\rho_{\text{hard}} - \rho_{\text{soft}}} \log^{3/2}\left(\frac{\sqrt{M}\gamma}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log \log\left(\frac{\sqrt{M}\gamma}{\rho_{\text{hard}} - \rho_{\text{soft}}}\right) \log\left(\frac{N_{\text{temp}}M}{\delta}\right)\right] \tag{B11}$$

calls to O_ρ . The number of calls to O_{Re} and O_{Im} is of the same order, since O_ρ contains $O(1)$ calls to them. Combining these observations, we obtain the estimations given by Eqs. (40) and (41) for the number of calls to O_{Re} and O_{Im} in the algorithm.

The number of qubits used in this algorithm is dominated by that required to perform O_{AE} in Eq. (45), since QAA does not require additional qubits. The number of qubits used for the operation in Eq. (45) is estimated as follows. The first two registers have

$$O(\log N_{\text{temp}} + \log M) \tag{B12}$$

qubits in total. Besides, according to Theorem 5, the third register and ancillary registers used for $O_{\rho,\epsilon,\delta',\sigma}^{\text{mean}}$ at the second arrow have

$$O\left\{\left[\log M + \log\left(\frac{\sigma}{\epsilon}\right)\right] \log\left(\frac{\sigma}{\epsilon}\right) \log \log\left(\frac{\sigma}{\epsilon}\right) \log\left(\frac{1}{\delta'}\right)\right\} \tag{B13}$$

qubits in total. Eq. (B13) becomes Eq. (39) under the setting on ϵ, σ , and δ' in Algorithm 3. Since Eq. (B12) is subdominant to Eq. (39), we have an upper bound on the total qubit number as Eq. (39).

(ii) $r(\rho_{\text{soft}}) = 0$

If $\text{QAA}(O_{\text{AE}}, \frac{1}{2N_{\text{temp}}M}, \frac{\delta}{2})$ outputs “failure”, the algorithm outputs “there is no signal”. In case QAA outputs some quantum state $|\psi\rangle$ by error, the algorithm goes to the second step where $\rho_{m,j}$ is classically calculated for $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$ given by the first two registers in $|\psi\rangle$. The output of the classical computation should be smaller than ρ_{soft} , since we are now considering the case of $r(\rho_{\text{soft}}) = 0$, which means that $\rho_{m,j} < \rho_{\text{soft}}$ holds for any $(m, j) \in [N_{\text{temp}}]_0 \times [M]_0$. Accordingly, the algorithm outputs “there is no signal” at the second step. In summary, at any rate, the algorithm outputs this message, if $r(\rho_{\text{soft}}) = 0$.

According to Theorem 2, in any cases, the number of calls to O_{AE} in $\text{QAA}(O_{\text{AE}}, \frac{1}{2N_{\text{temp}}M}, \frac{\delta}{2})$ is at most $O(\sqrt{N_{\text{temp}}M} \log \delta^{-1})$. As stated above, the number of calls to O_{Re} and O_{Im} in O_{AE} is given by Eq. (B11). Combining these, we obtain the bound Eq. (42) for the number of queries to O_{Re} and O_{Im} in the algorithm.

Since $\text{QAA}(O_{\text{AE}}, \frac{1}{2N_{\text{temp}}M}, \frac{\delta}{2})$ runs on the same system in any cases, the discussion on qubit number is the same as Case (i).

(iii) $r(\rho_{\text{hard}}) = 0$ and $r(\rho_{\text{soft}}) > 0$

Since the evaluation on the maximum number of calls to O_{Re} and O_{Im} obtained in Case (ii) also applies to this case, we have the same query complexity bound of Eq. (42). The discussion on qubit number is also the same as Case (ii). ■

[1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2002).

[2] L. Lin, Lecture notes on quantum algorithms for scientific computation, [arXiv:2201.08309](https://arxiv.org/abs/2201.08309).

[3] S. Gao, F. Hayes, S. Croke, C. Messenger, and J. Veitch, Quantum algorithm for gravitational-wave matched filtering, *Phys. Rev. Research* 4, 023006 (2022).

[4] R. Balasubramanian, B. S. Sathyaprakash, and S. V. Dhurandhar, Gravitational waves from coalescing binaries: Detection strategies and monte carlo estimation of parameters, *Phys. Rev. D* 53, 3033 (1996).

[5] B. J. Owen, Search templates for gravitational waves from inspiraling binaries: Choice of template spacing, *Phys. Rev. D* 53, 6749 (1996).

- [6] B. J. Owen and B. S. Sathyaprakash, Matched filtering of gravitational waves from inspiraling compact binaries: Computational cost and template placement, *Phys. Rev. D* **60**, 022002 (1999).
- [7] B. Allen, W. G. Anderson, P. R. Brady, D. A. Brown, and J. D. E. Creighton, FINDCHIRP: An algorithm for detection of gravitational waves from inspiraling compact binaries, *Phys. Rev. D* **85**, 122006 (2012).
- [8] B. P. Abbott *et al.*, GW150914: The Advanced LIGO Detectors in the Era of First Discoveries, *Phys. Rev. Lett.* **116**, 131103 (2016).
- [9] B. P. Abbott *et al.*, GWTC-1: A Gravitational-Wave Transient Catalog of Compact Binary Mergers Observed by LIGO and Virgo during the First and Second Observing Runs, *Phys. Rev. X* **9**, 031040 (2019).
- [10] R. Abbott *et al.*, GWTC-2: Compact Binary Coalescences Observed by LIGO and Virgo during the First Half of the Third Observing Run, *Phys. Rev. X* **11**, 021053 (2021).
- [11] R. Abbott *et al.*, GWTC-3: Compact binary coalescences observed by LIGO and Virgo during the second part of the third observing run, [arXiv:2111.03606](https://arxiv.org/abs/2111.03606).
- [12] L. K. Grover, A fast quantum mechanical algorithm for database search, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing* (ACM, New York, 1996), pp. 212–219.
- [13] J. W. Cooley, P. A. W. Lewis, and P. D. Welch, Historical notes on the fast fourier transform, *Proc. IEEE* **55**, 1675 (1967).
- [14] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, *Phys. Rev. A* **86**, 032324 (2012).
- [15] A. Montanaro, Quantum speedup of monte carlo methods, *Proc. R. Soc. A* **471**, 20150301 (2015).
- [16] B. Allen, χ^2 time-frequency discriminator for gravitational wave detection, *Phys. Rev. D* **71**, 062001 (2005).
- [17] C. Talbot and E. Thrane, Gravitational-wave astronomy with an uncertain noise power spectral density, *Phys. Rev. Research* **2**, 043298 (2020).
- [18] V. Vedral, A. Barenco, and A. Ekert, Quantum networks for elementary arithmetic operations, *Phys. Rev. A* **54**, 147 (1996).
- [19] D. Beckman, A. N. Chari, S. Devabhaktuni, and J. Preskill, Efficient networks for quantum factoring, *Phys. Rev. A* **54**, 1034 (1996).
- [20] T. G. Draper, Addition on a quantum computer, [arXiv:quant-ph/0008033](https://arxiv.org/abs/quant-ph/0008033).
- [21] S. A. Cuccaro, T. G. Draper, S. A. Kutin, and D. P. Moulton, A new quantum ripple-carry addition circuit, [arXiv:quant-ph/0410184](https://arxiv.org/abs/quant-ph/0410184).
- [22] Y. Takahashi and N. Kunihiro, A linear-size quantum circuit for addition with no ancillary qubits, *Quantum Inf. Comput.* **5**, 440 (2005).
- [23] T. G. Draper, S. A. Kutin, E. M. Rains, and K. M. Svore, A logarithmic-depth quantum carry-lookahead adder, *Quantum Inf. Comput.* **6**, 351 (2006).
- [24] J. J. Álvarez-Sánchez, J. V. Álvarez-Bravo, and L. M. Nieto, A quantum architecture for multiplying signed integers, *J. Phys.: Conf. Ser.* **128**, 012013 (2008).
- [25] Y. Takahashi and N. Kunihiro, A fast quantum circuit for addition with few qubits, *Quantum Inf. Comput.* **8**, 636 (2008).
- [26] Y. Takahashi, S. Tani, and N. Kunihiro, Quantum addition circuits and unbounded fan-out, *Quantum Inf. Comput.* **10**, 872 (2010).
- [27] A. Khosropour, H. Aghababa, and B. Forouzandeh, Quantum division circuit based on restoring division algorithm, in *2011 Eighth International Conference on Information Technology: New Generations* (IEEE, Piscataway, NJ, 2011), pp. 1037–1040.
- [28] L. Jamal and H. M. H. Babu, Efficient approaches to design a reversible floating point divider, in *2013 IEEE International Symposium on Circuits and Systems (ISCAS)* (IEEE, Piscataway, NJ, 2013), pp. 3004–3007.
- [29] H. V. Jayashree, H. Thapliyal, H. R. Arabnia, and V. K. Agrawal, Ancilla-input and garbage-output optimized design of a reversible quantum integer multiplier, *J. Supercomput.* **72**, 1477 (2016).
- [30] S. V. Dibbo, H. M. H. Babu, and L. Jamal, An efficient design technique of a quantum divider circuit, in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)* (IEEE, Piscataway, NJ, 2016), pp. 2102–2105.
- [31] H. M. H. Babu, Cost-efficient design of a quantum multiplier-accumulator unit, *Quant. Info. Proc.* **16**, 30 (2017).
- [32] E. Muñoz-Coreas and H. Thapliyal, Quantum circuit design of a T-count optimized integer multiplier, *IEEE Trans. Comput.* **68**, 729 (2018).
- [33] H. Thapliyal, E. Muñoz-Coreas, T. S. S. Varun, T. S. Humble, Quantum circuit designs of integer division optimizing T-count and T-depth, *IEEE Transactions on Emerging Topics in Computing* **9**, 1045 (2019).
- [34] T. Häner, M. Roetteler, and K. M. Svore, Optimizing quantum circuits for arithmetic, [arXiv:1805.12445](https://arxiv.org/abs/1805.12445).
- [35] I. Koren, *Computer Arithmetic Algorithms* (CRC Press, Boca Raton, FL, 2001).
- [36] D. J. Egger, R. G. Gutiérrez, J. C.é Mestre, and S. Woerner, Credit risk analysis using quantum computers, *IEEE Trans. Comput.* **70**, 2136 (2020).
- [37] L. Grover and T. Rudolph, Creating superpositions that correspond to efficiently integrable probability distributions, [arXiv:quant-ph/0208112](https://arxiv.org/abs/quant-ph/0208112).
- [38] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes, 3rd edition: The Art of Scientific Computing* (Cambridge University Press, Cambridge, 2007).
- [39] G. Brassard, P. Høyer, and A. Tapp, Quantum counting, in *International Colloquium on Automata, Languages, and Programming* (Springer, New York, 1998), pp. 820–831.
- [40] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, *Contemp. Math.* **305**, 53 (2002).
- [41] Y. Suzuki, S. Uno, R. Raymond, T. Tanaka, T. Onodera, and N. Yamamoto, Amplitude estimation without phase estimation, *Quant. Info. Proc.* **19**, 75 (2020).
- [42] S. Aaronson and P. Rall, Quantum approximate counting, simplified, in *Symposium on Simplicity in Algorithms* (SIAM, Philadelphia, 2020), pp. 24–32.
- [43] K. Nakaji, Faster amplitude estimation, *Quantum Inf. Comput.* **20**, 1109 (2020).
- [44] D. Grinko, J. Gacon, C. Zoufal, and S. Woerner, Iterative quantum amplitude estimation, *npj Quantum Inf.* **7**, 52 (2021).

- [45] T. Tanaka, Y. Suzuki, S. Uno, R. Raymond, T. Onodera, and N. Yamamoto, Amplitude estimation via maximum likelihood on noisy quantum computer, *Quant. Info. Proc.* **20**, 293 (2021).
- [46] S. Uno, Y. Suzuki, K. Hisanaga, R. Raymond, T. Tanaka, T. Onodera, and N. Yamamoto, Modified Grover operator for quantum amplitude estimation, *New J. Phys.* **23**, 083031 (2021).
- [47] T. Giurgica-Tiron, I. Kerenidis, F. Labib, A. Prakash, and W. Zeng, Low depth algorithms for quantum amplitude estimation, *Quantum* **6**, 745 (2022).
- [48] G. Wang, D. E. Koh, P. D. Johnson, and Y. Cao, Minimizing Estimation Runtime on Noisy Quantum Computers, *PRX Quantum* **2**, 010346 (2021).
- [49] T. Tanaka, S. Uno, T. Onodera, N. Yamamoto, and Y. Suzuki, Noisy quantum amplitude estimation without noise estimation, *Phys. Rev. A* **105**, 012411 (2022).
- [50] T. Giurgica-Tiron, S. Johri, I. Kerenidis, J. Nguyen, N. Piseni, A. Prakash, K. Sosnova, K. Wright, and W. Zeng, Low depth amplitude estimation on a trapped ion quantum computer, *Phys. Rev. Research* **4**, 033034 (2022).
- [51] M. R. Jerrum, L. G. Valiant, and V. V. Vazirani, Random generation of combinatorial structures from a uniform distribution, *Theor. Comput. Sci.* **43**, 169 (1986).
- [52] R. Asaka, K. Sakai, and R. Yahagi, Quantum circuit for the fast fourier transform, *Quant. Info. Proc.* **19**, 277 (2020).
- [53] V. Giovannetti, S. Lloyd, and L. Maccone, Quantum Random Access Memory, *Phys. Rev. Lett.* **100**, 160501 (2008).
- [54] C. Cutler *et al.*, The Last Three Minutes: Issues in Gravitational Wave Measurements of Coalescing Compact Binaries, *Phys. Rev. Lett.* **70**, 2984 (1993).
- [55] M. Maggiore, *Gravitational Waves. Vol. 1: Theory and Experiments*, Oxford Master Series in Physics (Oxford University Press, Oxford, 2007).
- [56] S. Khan, K. Chatziioannou, M. Hannam, and F. Ohme, Phenomenological model for the gravitational-wave signal from precessing binary black holes with two-spin effects, *Phys. Rev. D* **100**, 024059 (2019).
- [57] J. Aasi *et al.*, Advanced LIGO, *Class. Quantum Grav.* **32**, 115012 (2015).
- [58] P. Ajith, M. Hannam, S. Husa, Y. Chen, B. Brügmann, N. Dorband, D. Müller, F. Ohme, D. Pollney, C. Reisswig, L. Santamaría, and J. Seiler, Inspiral-Merger-Ringdown Waveforms for Black-Hole Binaries with Nonprecessing Spins, *Phys. Rev. Lett.* **106**, 241101 (2011).
- [59] P. D. Lasky, Gravitational waves from neutron stars: A review, *Publ. Astron. Soc. Austral.* **32**, e034 (2015).
- [60] K. Glampedakis and L. Gualtieri, Gravitational waves from single neutron stars: An advanced detector era survey, *Astrophys. Space Sci. Libr.* **457**, 673 (2018).
- [61] K. Riles, Recent searches for continuous gravitational waves, *Mod. Phys. Lett. A* **32**, 1730035 (2017).
- [62] M. Sieniawska and M. Bejger, Continuous gravitational waves from neutron stars: Current status and prospects, *Universe* **5**, 217 (2019).
- [63] R. Tenorio, D. Keitel, and A. M. Sintes, Search methods for continuous gravitational-wave signals from unknown sources in the advanced-detector era, *Universe* **7**, 474 (2021).
- [64] S. Arunachalam, V. Gheorghiu, T. Jochym-O'Connor, M. Mosca, and P. V. Srinivasan, On the robustness of bucket brigade quantum RAM, *New J. Phys.* **17**, 123010 (2015).