

Quantum period finding using one-qubit reduced density matrices

Marco Bernardi ^{*}

*Department of Applied Physics and Materials Science, and Department of Physics,
California Institute of Technology, Pasadena, California 91125, USA*



(Received 17 November 2025; accepted 13 April 2026; published 8 May 2026)

The quantum period-finding (QPF) algorithm can compute the period of a function exponentially faster than the best-known classical algorithm. In standard QPF, the output state has a primary contribution from r high-probability bit strings, where r is the period. Measurement of this state, combined with continued fraction analysis, reveals the unknown period. Here, we explore a different approach to QPF, where the period is obtained from single-qubit quantities—specifically, the set of one-qubit reduced density matrices (1-RDMs)—rather than the output bit strings of the entire quantum circuit. Using state-vector simulations, we compute the 1-RDMs of the QPF circuit for a generic periodic function. Analysis of these 1-RDMs as a function of period reveals distinctive patterns, which allows us to obtain the unknown period from the 1-RDMs using a numerical root-finding approach. Our results show that the 1-RDMs—a set of $\mathcal{O}(n)$ one-qubit marginals—contain enough information to reconstruct the period, which is typically obtained by sampling the space of $\mathcal{O}(2^n)$ bit strings. Conceptually, this can be viewed as a “compression” of the information in the QPF algorithm, which enables period finding from n one-qubit marginals. Our results motivate the development of approximate simulations of reduced density matrices to design period-finding algorithms.

DOI: [10.1103/bcxp-1qld](https://doi.org/10.1103/bcxp-1qld)

I. INTRODUCTION

Among quantum algorithms envisioned to achieve an advantage over classical computers, Shor’s algorithm for integer factorization [1,2] provides a compelling example of how quantum computers could dramatically speed up calculations for useful tasks [3–9]. In Shor’s algorithm, the factorization of a semiprime integer is turned into a period-finding problem [10,11], where factoring an n -bit semiprime integer $S < N$, with $N = 2^n$, requires finding the period of the modular exponential function, $a^x \bmod S$, where a and S are relative primes. Finding this period requires $\mathcal{O}(2^n)$ resources on a classical computer: Since the period r can take any value from 1 to $N - 1$, a brute-force approach testing the possible periods would require on average $N/2 = 2^{n-1}$ attempts [12].

In contrast, on a quantum computer, one can use the quantum period-finding (QPF) algorithm, which can find the period using resources that scale polynomially with the number of bits n of the integer to be factored, achieving an exponential speedup over its classical counterpart. The final step of the QPF algorithm measures the quantum state generated by the quantum Fourier transform (QFT). This state has a dominant component in a subspace spanned by r high-probability bit strings, where r is the period, as a result of

interference and quantum correlations that are essential to achieve quantum speedup [10]. The QPF algorithm can be accurately modeled on a classical computer using state-vector simulations, but because of the $\mathcal{O}(2^n)$ size of the Hilbert space, such state-vector simulations perform similarly to the classical algorithm based on brute-force period testing and are currently limited to ~ 50 qubits with state-of-the-art classical computers [13–16].

Approximate methods for classical simulation of quantum circuits (QCs) provide an alternative to exact state-vector simulations. Examples include QC simulations based on tensor networks [17–19], density matrices [20,21], and neural-network quantum states [22,23], among others [24–26]. In particular, reduced density matrices (RDMs) can capture multi-qubit correlations and are smaller objects than state vectors: The combined size of the s -qubit RDM (s -RDM) matrices is $\mathcal{O}(n^s)$ for a QC with n qubits. These RDMs can be computed exactly starting from the state vector, with exponential cost $\mathcal{O}(2^n)$, while approximate RDMs can be obtained at lower cost using approaches based on open system dynamics [27–29]. Recent work has proposed efficient schemes for approximate simulations of one-qubit RDMs (1-RDMs), obtaining accurate results for specific random QCs and oracle-based QCs [30]. However, the structure of exact 1-RDMs (and higher s -RDMs) is not known for most quantum algorithms. This knowledge could shed light on quantum correlations and advance classical simulations of QCs.

Here, we study the exact 1-RDMs of the QPF algorithm and discover a regular peak structure as a function of qubit and period. Knowledge of this pattern allows us to derive an approximate expression for the diagonal elements of the 1-RDMs, as a function of period for each qubit, valid for

^{*}Contact author: bmarco@caltech.edu

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.

a generic periodic function. We show how to find the unknown period using a root-finding algorithm that compares the computed and approximate 1-RDMs. Similar to continued fractions in standard QPF, our method requires $2n$ qubits to find an n -bit period with unit accuracy. This approach reconstructs the period from n one-qubit marginals of the QC distribution and is conceptually different from standard QPF, where the period is obtained by sampling bit strings in a subspace of approximate size r , which scales as 2^n for a generic period. This work focuses on exact 1-RDMs, and thus finding the period still requires exponential resources. Yet, the approach developed here may inspire more efficient classical simulations of QPF and Shor's algorithm using approximate RDMs. We plan to investigate this topic in future work.

II. THEORY

A. Reduced density matrices

We study a periodic function $f(x) = f(x + r)$ defined for integers $x \in [0, 2^n - 1]$, where n is the number of qubits, $N = 2^n$ is the size of the domain, and the period r can range from 1 to $N - 1$. The 1-RDM for qubit q is a 2×2 matrix obtained by tracing out $n - 1$ qubits (all qubits k except q) from the QC density matrix, $|\Psi\rangle\langle\Psi|$, where $|\Psi\rangle$ is the QC state vector. The entire set of 1-RDMs, one per qubit, can be computed using

$$\rho_{ij}^{(q)} = \text{Tr}_{\{k\} \neq q}(|\Psi\rangle\langle\Psi|) = \sum_{\{b_k\} \neq b_q} \langle b_{n-1} \cdots b_q = i \cdots b_0 | \Psi \rangle \times \langle \Psi | b_{n-1} \cdots b_q = j \cdots b_0 \rangle, \quad (1)$$

where $|b\rangle = |b_{n-1} \cdots b_1 b_0\rangle \in \{0, 1\}^n$ are binary strings corresponding to integers $b = \sum_{l=0}^{n-1} b_l 2^l$ in $[0, 2^n - 1]$ (where $b_l = 0$ or 1). The indices i and j , equal to 0 or 1, label the elements of the 1-RDM matrices.

For our problem involving periodic functions, we study the structure of the 1-RDMs as a function of period r for each qubit q in the QC, focusing on $\rho^{(q)}(r)$ as the key quantities of interest. The 1-RDMs are Hermitian matrices with unit trace, so they can be written as

$$\rho^{(q)}(r) = \frac{1}{2} + \boldsymbol{\sigma} \cdot \mathbf{a}^{(q)}(r) = \begin{pmatrix} 0.5 + a_z & a_x - ia_y \\ a_x + ia_y & 0.5 - a_z \end{pmatrix}, \quad (2)$$

where $\mathbf{a}^{(q)}(r) = \{a_x, a_y, a_z\}$ are real-valued coefficients for each qubit and period, here satisfying $2|\mathbf{a}^{(q)}(r)| = 1$ (pure state), and $\boldsymbol{\sigma} = \{\sigma_x, \sigma_y, \sigma_z\}$ are Pauli matrices. The coefficients in the diagonal elements of the 1-RDMs, $a_z^{(q)}(r)$, are analyzed in detail in this work.

B. Quantum period-finding algorithm

In the QPF circuit (see Fig. 1) [10,11], the periodic function f is prepared using two registers, each with n qubits. After preparing the superposition state $H^{\otimes n}|0\rangle$ in the first register, where H is the Hadamard gate, an oracle prepares the function f in the second register. Following a measurement of the function in the second register, which returns $f(a_0)$, the state

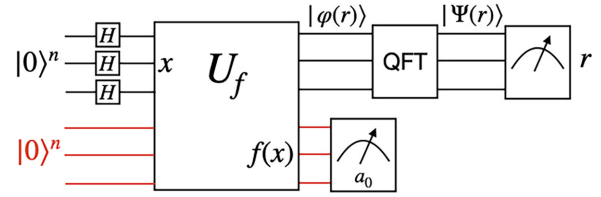


FIG. 1. Quantum circuit for QPF, shown for $n = 3$ qubits in both registers. The oracle prepares the periodic function f in the second register, shown at the bottom in red. Measuring the second register puts the first register in state $|\varphi(r)\rangle$, which becomes $|\Psi(r)\rangle$ after the quantum Fourier transform (see text for definitions). The final measurement, combined with continued fraction analysis, provides the period r .

in the first register becomes [11]

$$|\varphi(r)\rangle = \frac{1}{\sqrt{\lceil N/r \rceil}} \sum_{m=0}^{\lceil N/r \rceil} |a_0 + mr\rangle, \quad (3)$$

where $\lfloor N/r \rfloor$ and $\lceil N/r \rceil$ are, respectively, the floor and ceiling of the ratio N/r . Applying the quantum Fourier transform on the first register generates the output state

$$|\Psi(r)\rangle = \sum_{b=0}^{N-1} \left(\frac{1}{\sqrt{N \lceil N/r \rceil}} \sum_{m=0}^{\lceil N/r \rceil} e^{i2\pi (a_0 + mr) \frac{b}{N}} \right) |b\rangle. \quad (4)$$

(Above, when N/r is an integer, $\lceil N/r \rceil$ should be replaced by $N/r - 1$.) Measurement of this state in the computational basis, combined with continued-fraction analysis [10,11], provides the unknown period r . Writing Eq. (4) as $|\Psi(r)\rangle = \sum_b C_b(r) |b\rangle$, it is clear that the final measurement will return a bit string b with probability $|C_b(r)|^2$, which is independent of the value $f(a_0)$ measured in the second register [11].

C. 1-RDMs for quantum period finding

The diagonal elements of the 1-RDM play a central role in our treatment. Using Eq. (1) with the QC output state $|\Psi(r)\rangle$ defined in Eq. (4), we obtain the diagonal elements as

$$\begin{aligned} \rho_{00}^{(q)}(r) &= \sum_{\{b_k\} \neq b_q} |\langle b_{n-1} \cdots b_q = 0 \cdots b_0 | \Psi \rangle|^2 \\ &= \sum_{b \in \beta_q} \frac{1}{N \lceil N/r \rceil} \left| \sum_{m=0}^{\lceil N/r \rceil} \exp\left(i2\pi m \frac{rb}{N}\right) \right|^2 \\ &= \sum_{b \in \beta_q} |C_b(r)|^2, \end{aligned} \quad (5)$$

where we define β_q as the set of binary strings with the digit b_q equal to zero and all other digits equal to 0 or 1. In the last line, the coefficient $|C_b(r)|^2$ is the probability of measuring a bit string b as the output of the QC and is a familiar quantity in Shor's algorithm [10,11]. For a generic period r , $|C_b(r)|^2$ is maximal when rb/N is close to an integer, and thus for “special” high-probability strings b^* satisfying $b^* \approx j(N/r)$, with $j = 0, 1, \dots, r - 1$. This becomes an exact equality when the period is a divisor of the domain size $N = 2^n$, and thus for

$r = 2^k$ with $0 \leq k < n$. Note also that $\rho_{00}^{(q)}(r)$ is independent of the value $f(a_0)$ measured in the second register.

In contrast, the coherences $\rho_{01}^{(q)}(r)$ depend on a_0 and are therefore less important to find patterns in the 1-RDMs. This is seen by using the 1-RDM definition in Eq. (1):

$$\rho_{01}^{(q)}(r) = \sum_{\{b_k\} \neq b_q} \langle b_{n-1} \cdots b_q = 0 \cdots b_0 | \Psi \rangle \times \langle \Psi | b_{n-1} \cdots b_q = 1 \cdots b_0 \rangle, \quad (6)$$

from which one finds that the coherences involve two terms where the coefficients C_b are summed over different sets of binary strings:

$$\rho_{01}^{(q)}(r) = \sum_{b \in \beta_q} \frac{1}{N \lceil N/r \rceil} \left(\sum_{m=0}^{\lfloor N/r \rfloor} \exp \left(i2\pi (a_0 + mr) \frac{b}{N} \right) \right) \times \left(\sum_{m=0}^{\lfloor N/r \rfloor} \exp \left(-i2\pi (a_0 + mr) \frac{\xi(b)}{N} \right) \right). \quad (7)$$

In the first term in parentheses, the summation runs over $b \in \beta_q$, the set of strings with digit 0 for qubit q , while the second term contains the complementary strings $\xi(b)$, which are identical to the respective strings b but with the digit at qubit q set equal to 1. Because the two terms sum over different strings, the phase factor containing a_0 does not cancel and $\rho_{01}^{(q)}(r)$ depends on a_0 . Our numerical tests (not discussed here) confirm that $\rho_{01}^{(q)}(r)$ oscillates rapidly as a function of period without showing a clear pattern.

For this reason, in this work we focus on the diagonal elements $\rho_{00}^{(q)}(r)$, whose physical meaning is clear from Eq. (5): $\rho_{00}^{(q)}(r)$ is the probability of measuring zero for a single-qubit measurement on qubit q in the output of the QPF circuit and can be written as the sum of probabilities $|C_b(r)|^2$ of measuring a bit string $b \in \beta_q$. Therefore, $\rho_{00}^{(q)}(r)$ is a single-qubit probability, also known as a one-qubit marginal. This quantity has been studied in recent mean-field simulations of QCs [30].

III. RESULTS

In Secs. III A and III B, we show calculations of 1-RDMs for the QPF circuit, focusing on the diagonal elements $\rho_{00}^{(q)}(r)$. Section III A discusses the simple case where the period r is a divisor of the domain size $N = 2^n$. This corresponds to the case where N/r is an integer, so the period equals $r = 2^k$ with $0 \leq k < n$, and the summations in Eq. (5) can be evaluated analytically. Section III B focuses on the remaining periods that are not divisors of the domain size. In this case, N/r is not an integer and the 1-RDMs need to be computed numerically. In Sec. III C, we analyze the calculated $\rho_{00}^{(q)}(r)$ to identify patterns as a function of period and qubit number. This allows us to derive an approximate expression for $\rho_{00}^{(q)}(r)$, valid for a generic periodic function, and develop a numerical algorithm to obtain the period from the 1-RDMs. Finally, in Sec. III D, we explore the effects of noise on the proposed period-finding method.

A. 1-RDMs for quantum period finding: case $r = 2^k$

For periods $r = 2^k$, we can derive analytical expressions for the 1-RDMs because N/r is an integer. We first evaluate the sum over m in Eq. (5):

$$\rho_{00}^{(q)}(r = 2^k) = \sum_{b \in \beta_q} \frac{1}{N \times N/r} \left| \sum_{m=0}^{N/r-1} (e^{i2\pi \frac{rb}{N}})^m \right|^2 = \frac{1}{N \times N/r} \sum_{b \in \beta_q} \left(\frac{1 - e^{i2\pi \frac{rb}{N}}}{1 - e^{i2\pi \frac{rb}{N}}} \right)^2, \quad (8)$$

We now need to evaluate the sum over $b \in \beta_q$ for each qubit q in the first register of the QPF circuit. In the second line of Eq. (8), the numerator in parentheses is always zero because b is an integer, and the only terms contributing to the sum are those where the denominator is also zero, which requires that rb/N is an integer. When this condition is satisfied, using L'Hopital's rule, each such term contributes $(N/r)^2$ to the sum [11].

Let us count how many terms have an integer ratio rb/N . Using $r = 2^k$ and $N = 2^n$, where n is the number of qubits and $k \in [0, n-1]$, we write the ratio as

$$\frac{rb}{N} = \sum_{l=0}^{n-1} b_l 2^{l+k-n}, \quad (9)$$

where we expressed b in binary notation, $b = \sum_{l=0}^{n-1} b_l 2^l$. From Eq. (9), the ratio rb/N is an integer only when all exponents $l+k-n \geq 0$, and thus for bit strings with

$$b_l = 0 \quad \text{for } l < n-k. \quad (10)$$

There can be up to $2^k = r$ such strings because the last k digits, b_l with $l \geq n-k$, can take any value.

Depending on the qubit q being considered, the number of strings with integer ratio rb/N , for $b \in \beta_q$, can be equal to this maximum allowed value of 2^k , or smaller than that. To count such strings for qubit q , we examine the binary strings in the set β_q , which have digit $b_q = 0$.

If $q < n-k$, Eq. (10) is satisfied, and thus there are $r = 2^k$ nonzero terms in the sum over β_q in Eq. (8), each contributing $(N/r)^2$. Therefore, we obtain

$$\rho_{00}^{(q)}(r = 2^k) = \frac{r}{N^2} \times r \left(\frac{N}{r} \right)^2 = 1 \quad \text{for } q < n-k. \quad (11)$$

If $q \geq n-k$, the set β_q contains only half of the 2^k strings satisfying Eq. (10), because $b_q = 0$ for $q \geq n-k$. This leads to a total of $2^{k-1} = r/2$ nonzero terms in the sum over β_q in Eq. (8), and thus we obtain

$$\rho_{00}^{(q)}(r = 2^k) = \frac{r}{N^2} \times \frac{r}{2} \left(\frac{N}{r} \right)^2 = 0.5 \quad \text{for } q \geq n-k. \quad (12)$$

In the following, we express the 1-RDMs in terms of the coefficients a defined in Eq. (2). In particular, a_z is related to the diagonal elements of the 1-RDMs via $\rho_{00} = 0.5 + a_z$. To study $a_z^{(q)}(r)$ as a function of period r and qubit q , it is useful

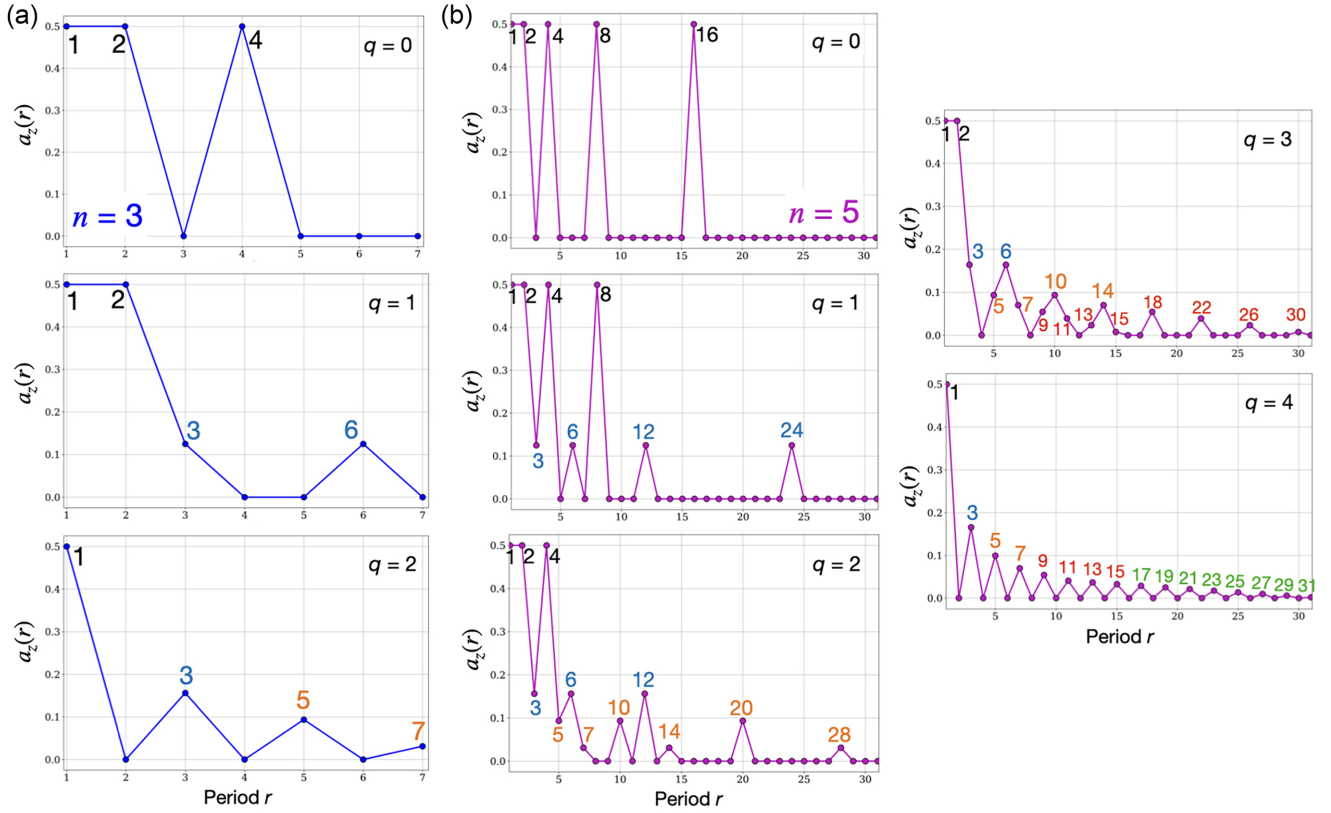


FIG. 2. Computed diagonal elements of the 1-RDMs for the QPF circuit, expressed as $a_z^{(q)}(r) = \rho_{00}^{(q)}(r) - 0.5$, shown as a function of period r for each qubit q . We show results for (a) $n = 3$ and (b) $n = 5$ qubits. The periods with $a_z^{(q)}(r) > 0$ fit the general rule in Eq. (14). For qubit $q = 0$, a nonzero value of $a_z^{(q)}(r) > 0$ occurs only for periods $r = 2^k$. For each subsequent qubit ($q = 1, 2, \dots, n - 1$), there are additional periods with $a_z^{(q)}(r) > 0$, shown with colors not used in previous qubits, while some periods that had $a_z > 0$ in preceding qubits are removed. The last qubit, $q = n - 1$, has $a_z(r) > 0$ only for odd periods.

to summarize the results above for $r = 2^k$ as

$$\begin{aligned} a_z^{(q)}(r = 2^k) &= 0.5 \quad \text{for } k = 0, 1, \dots, n - q - 1 \\ a_z^{(q)}(r = 2^k) &= 0 \quad \text{for } k = n - q, n - q + 1, \dots, n - 1. \end{aligned} \quad (13)$$

This is a special case of a more general pattern discussed below.

B. 1-RDMs for quantum period finding: case $r \neq 2^k$

When $r \neq 2^k$, the period is not a divisor of the domain size $N = 2^n$. Therefore, rb/N is not an integer, so the sums in Eq. (5) cannot be evaluated analytically and the 1-RDMs need to be computed numerically. We compute the exact 1-RDMs using state-vector simulations of the QPF circuit followed by taking partial traces. The calculations loop over all possible periods and provide $\rho_{00}^{(q)}(r)$ for all qubits. In our analysis, we are only interested in the 1-RDMs for the n qubits in the first register of the QPF circuit (see Fig. 1), the one where the QFT is applied, while the 1-RDMs for the second register are ignored. The code and datasets for all simulations and 1-RDM calculations are provided in the Supplemental Material [31].

To analyze the results, we focus on the physical meaning of $\rho_{00}^{(q)}(r)$. Since $\text{Tr}(\rho) = \rho_{00} + \rho_{11} = 1$, when there is no constructive or destructive interference in the sum over strings in β_q in Eq. (5), one expects an average value of

$\rho_{00}^{(q)}(r) = 0.5$, and thus $a_z^{(q)}(r) = 0$, because β_q includes only half of the strings. Therefore, we can regard the value $a_z = 0$ as the baseline, and any value $a_z^{(q)}(r) > 0$ as the result of constructive interference.

Analysis of our computed 1-RDMs provides a pattern for the nonzero values of $a_z^{(q)}(r)$, which includes the analytic results in Eq. (13) for $r = 2^k$ as a special case, and generalizes them to all periods. For $r = 2^k$, Eq. (13) shows that $a_z^{(q)}(r)$ exhibits a series of peaks for periods equal to certain powers of 2. In particular, for a given qubit q , the value $a_z^{(q)} = 0.5$ occurs for periods $r = 2^k$ with $k = 0, \dots, n - 1 - q$, while $a_z^{(q)} = 0$ for periods equal to higher powers of 2.

Using numerical results for the remaining periods, we are able to generalize this trend. Specifically, we find that $a_z^{(q)}(r) > 0$ only for certain periods:

$$\begin{aligned} a_z^{(q)}(r) &> 0 \quad \text{for } r = 2^k \times r', \quad \text{with odd } r' < 2^{q+1} \\ &\text{and } k = 0, \dots, n - 1 - q, \\ a_z^{(q)}(r) &= 0 \quad \text{for all other periods } r. \end{aligned} \quad (14)$$

To illustrate this trend, in Fig. 2(a) we plot the results for $n = 3$ qubits. For qubit $q = 0$, we find $a_z^{(q)}(r) = 0.5$ for $r = \{1, 2, 4\}$, in agreement with Eq. (14) for $r' = 1$. Qubit $q = 1$ introduces periods with $a_z^{(q)}(r) > 0$, including $r' = 3$ and that period multiplied by powers of 2, $2^k \times r'$, with $k = 0, \dots, n - 1 - q$, consistent with Eq. (14). Similarly, for qubit $q = 2$, we find

additional periods with nonzero a_z values, the periods $r' = \{5, 7\}$. Note that for the last qubit, $q = n - 1$, the nonzero values of a_z occur only for odd-valued periods, $r = 2^0 r'$, with odd $r' < 2^n$, in agreement with the general trend given in Eq. (14).

In Fig. 2(b), we show the same pattern for the more complex case with $n = 5$ qubits. Following our discussion above, a new set of periods with $a_z > 0$ will appear for each of the qubits $q = 0, 1, 2$. These periods can be written as $2^k \times r'$, with $r' = 1$ for qubit 0, $r' = 3$ for qubit 1, and $r' = \{5, 7\}$ for qubit 2. There are additional sets of periods with $a_z > 0$ for qubits $q = 3, 4$. These periods can also be written as $2^k \times r'$, with $r' = \{9, 11, 13, 15\}$ for qubit 3, and all odd periods in the second half of the domain, $r' = \{17, 19, \dots, 31\}$ for qubit 4, which is the last qubit. This leaves only odd periods with $a_z > 0$ for the last qubit, similar to the case with $n = 3$.

Based on these results, the trend, in terms of periods with $a_z > 0$, is that starting at $q = 0$ each following qubit will (1) add periods $2^k \times r'$, for $2^q < r' < 2^{q+1}$ (2^{q-1} new odd periods r' , and these periods multiplied by powers of 2); and (2) remove some periods with $a_z > 0$ in previous qubits: specifically, remove periods $2^{n-q} \times r'$, with $r' < 2^q$. We have verified that these trends are general and hold for larger circuit sizes up to $n > 10$. Although these results cannot be derived analytically since Eq. (5) needs numerical calculations for generic periods, we can rationalize their origin, as we explain in the following.

C. Finding the period from the 1-RDMs

To find the unknown period from the 1-RDMs, we are given a set of n values $a_z^{(q)}$, one for each qubit q . These values can be obtained from single-qubit measurements or classical simulations of the QPF circuit in Fig. 1. The key question is whether the diagonal elements of the 1-RDMs—the n values $a_z^{(q)}$, which are one-qubit marginals of the QC distribution—contain enough information to reconstruct the period. In the following, we show that these quantities are a unique fingerprint of the period. We then show how to obtain the period from the n values $a_z^{(q)}$ (1-RDM diagonal elements) using a numerical root-finding approach.

1. Numerical approach for quantum period finding from the 1-RDMs

Starting from the patterns discussed above, our strategy is to first derive an accurate approximation for the 1-RDM diagonal elements, $a_z^{(q)}(r)$, as a function of the period, and then compare the calculated $a_z^{(q)}$ with the approximate values to find the period. Let us denote the approximate 1-RDM diagonal elements as $\tilde{a}_z^{(q)}(r)$. To find the unknown period r from the values $a_z^{(q)}$, we propose to solve for r in the equation

$$a_z^{(q)} - \tilde{a}_z^{(q)}(r) = 0. \quad (15)$$

A possible approach would be to find the period that minimizes the distance between the calculated and approximate $a_z^{(q)}$. However, since the number of possible periods spans the entire domain, this period-finding approach would have an exponential computational cost of order $\mathcal{O}(2^n)$. To avoid this complexity, after deriving approximate $\tilde{a}_z^{(q)}(r)$, we treat Eq. (15) as a root-finding problem and use the secant method

[32] to solve numerically for the period r . The numerical approach is described in detail in the following.

The key to our method is the derivation of approximate 1-RDMs, $\tilde{a}_z^{(q)}(r)$. From the definition in Eq. (5), $\rho_{00}^{(q)}(r) = \sum_{b \in \beta_q} |C_b(r)|^2$, we know that $\rho_{00}^{(q)}(r)$ is a sum of contributions from strings $b \in \beta_q$ that are approximately equal to an integer multiple of N/r . For the case $r = 2^k$, this becomes an exact equality and $a_z^{(q)}(r)$ can be computed analytically, as discussed above. For a generic period r , we approximate $\rho_{00}^{(q)}(r)$ by counting the number of integer multiples of N/r that fall within the range of values covered by the strings in β_q :

$$\rho_{00}^{(q)}(r) \approx \frac{1}{r} \sum_{j=0}^{r-1} \delta_{jN/r, b \in \mathcal{R}(\beta_q)}, \quad (16)$$

where $\mathcal{R}(\beta_q)$ is the range of real values spanned by the strings in β_q , which consists of a set of intervals on the real number line (see below). This formula counts the number of values $b \in \mathcal{R}(\beta_q)$ equal to jN/r , where $j = 0, \dots, r-1$, and normalizes the result by r . When the number of matching values is greater than $r/2$, we expect $\rho_{00} > 1/2$, and thus $a_z > 0$. This shows that a value of $a_z > 0$ is associated with constructive interference.

Counting the matching values in Eq. (16), we obtain an approximate formula (see the derivation in the next section): For qubit $n - 1 - q'$, namely, qubit q' from the last, the approximate 1-RDM diagonal elements for the periods $r = 2^{q'} \times r'$, with odd $r' < N/2^{q'}$, read

$$\tilde{\rho}_{00}^{(n-1-q')}(r) \approx \frac{2^{q'}}{r} \lceil (r/2^{q'})/2 \rceil, \quad (17)$$

where $\lceil x \rceil$ denotes the ceiling of x . Using $a_z = \rho_{00} - 0.5$, we obtain the approximate values $\tilde{a}_z^{(q)}(r)$ to be used in Eq. (15):

$$\tilde{a}_z^{(n-1-q')}(r) \approx \frac{2^{q'}}{r} \lceil (r/2^{q'})/2 \rceil - 0.5. \quad (18)$$

Figure 3 compares this approximate formula with the exact $a_z^{(q)}(r)$ from state-vector simulations for the last two qubits in the first register of the QPF circuit. For qubit $n - 1 - q'$, the approximate formula can accurately predict the local peaks of $a_z^{(q)}(r)$, which occur for periods $r = 2^{q'} \times r'$, with odd $r' < N/2^{q'}$. This is particularly true for periods in the first quarter of the domain, $r < N/4$, which allows us to find any period in a given domain by adding a few extra qubits to push the period of interest well into the first quarter. The accuracy of $\tilde{a}_z^{(q)}(r)$ in that region is the key to finding the period by solving Eq. (15) numerically.

We describe the algorithm to find the period from the 1-RDM diagonal elements. The corresponding code is provided in the notebook included in the Supplemental Material [31].

(1) We first generate a list of periods compatible with the values of $a_z^{(q)}$. Starting at qubit 0, we find the first qubit where $a_z^{(q)} > 0$. If this occurs at qubit l , then the unknown period has the form $2^k \times r'$, with odd r' such that $r' \in [2^l, 2^{l+1})$, for some value of k .

(2) We then find the value of k to restrict the list of possible periods. Starting at the last qubit, we find the first qubit from the last where $a_z^{(q)} > 0$. If this occurs for qubit q' from the last,

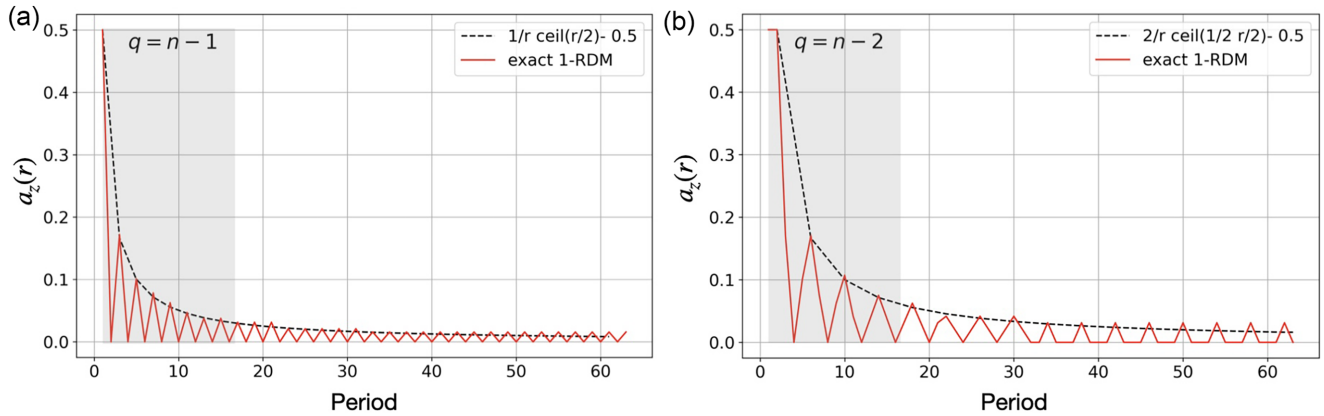


FIG. 3. Comparison of approximate vs exact $a_z^{(q)}(r)$ for $n=6$ qubits, shown for (a) the last qubit, $q = n - 1$, and (b) the penultimate qubit, $q = n - 2$. These cases correspond to $q'=0$ and $q'=1$, respectively. The first quarter of the domain (periods smaller than $N/4$), where the approximation is more accurate, is shown with a shaded area in both plots.

defined as qubit $q^* = n - 1 - q'$, then we can restrict the list of possible periods to $2^{q'} \times r'$, for odd $r' \in [2^l, 2^{l+1})$.

(3) If the period is in the sets 2^k or $2^k \times 3$, then Steps 1 and 2 are sufficient to find the period. For other cases, we need to refine our search within the list of possible periods generated in the previous step.

(4) For this refinement, we examine $a_z^{(q^*)}$ for qubit q^* defined above. Using Eqs. (15) and (18), we solve for r in $a_z^{(q^*)} - \tilde{a}_z^{(q^*)}(r) = 0$ using the secant method [32] over the set of possible periods found in Step 2, starting at the midpoint of that range of periods.

(5) To validate the result, we verify that the period found with the secant method is in the set of possible periods from Step 2. As the procedure is more accurate when $r < N/4$, we then rerun the period-finding algorithm by adding more qubits, until the computed period converges.

We use this approach to find periods in domains of integers with $n = 6, 7$, and 8 bits, respectively (periods up to $N = 256$). Although only n qubits are strictly necessary to find periods up to 2^n , the use of additional qubits pushes the periods of interest to the early part of the domain, where our approximation for $a_z^{(q)}(r)$ is more accurate (see Fig. 3). In Fig. 4(a), we plot the accuracy of our method, defined as the fraction of periods in the domain predicted correctly, as a function of extra qubits. For n -bit integers, we find that using n extra qubits, for a total of $2n$ qubits, allows us to predict the period with unit accuracy for all the domain sizes studied here.

Figure 4(b) shows the distribution of errors within the domain for the case of $n = 7$ qubits ($N = 128$) for several values of extra qubits, n_{extra} , ranging from 2 to 6. For $n_{\text{extra}} = 2$, most periods in the first quarter of the domain are already predicted correctly, but many periods in the center and upper part of the domain are not. Increasing the number of extra qubits removes errors in the center of the domain and progressively increases the accuracy for the largest integers in the domain. Only a few periods are predicted incorrectly for $n_{\text{extra}} = 6$, and all periods are predicted correctly for $n_{\text{extra}} = 7$ (not shown). The requirement of using $2n$ qubits to find the period with unit accuracy from the 1-RDMs is equivalent to the standard QPF algorithm based on bit strings—for example, in Shor's

algorithm—which requires $2n$ qubits to find the period reliably using continued fractions [10].

2. Derivation of 1-RDM approximate formula

According to Eq. (16), $\rho_{00}^{(q)}(r) = a_z^{(q)}(r) + 0.5$ counts the number of strings $b \in \beta_q$ that are approximately equal to jN/r , with $j = 0, \dots, r - 1$. To derive the approximate formula for $\tilde{\rho}_{00}^{(q)}(r)$ in Eq. (17), we focus on β_q , the set of integers corresponding to binary strings of length n with digit q equal to 0, and compare these integers with jN/r .

We first analyze the last qubit, $q = n - 1$, where for the exact 1-RDMs, $a_z^{(n-1)}(r) > 0$ for odd periods and $a_z^{(n-1)}(r) = 0$ for even periods in the domain [see Fig. 3(a)]. We want to understand the origin of this trend to formulate approximate 1-RDMs. Writing the strings as $b = (b_{n-1}b_{n-2} \dots b_1b_0)$, we have $\beta_q = \{(0b_{n-2} \dots b_1b_0)\}$ for $q = n - 1$, which corresponds to integers that span the first half of the domain, the interval from 0 to $N/2 - 1$, with $N = 2^n$. We define this interval as the range of real values spanned by β_q and denote it by $\mathcal{R}(\beta_q)$.

For odd periods, here denoted as r' , more than half of the values jN/r' fall in the first half of the domain, which is spanned by $\mathcal{R}(\beta_q)$. For example, consider the case $r' = 7$, illustrated in Fig. 5(a). The values jN/r' , with $j = 0, \dots, r' - 1$, are $\{0, N/7, 2N/7, 3N/7, 4N/7, 5N/7, 6N/7\}$, and thus 4 out of 7 values of jN/r' are in the first half of the domain, where 4 equals the ceiling of $r'/2$. This shows that Eq. (17), for $q = n - 1$ (and thus $q' = 0$), holds in this case:

$$\tilde{\rho}_{00}^{(n-1)}(r') \approx \frac{1}{r'} \lceil r'/2 \rceil = \frac{4}{7} > 0.5, \quad r' \text{ odd}. \quad (19)$$

The result $\tilde{\rho}_{00}^{(n-1)}(r') = 1/r' \lceil r'/2 \rceil > 0.5$, and thus $a_z^{(n-1)}(r') > 0$, applies to all odd periods for the last qubit.

In contrast, for even periods r , exactly half of the values jN/r fall in the first half of the domain spanned by β_q . Consider the case $r = 6$, with $jN/r = \{0, N/6, N/3, N/2, 2N/3, 5N/6\}$. Only 3 of the 6 values are in the range from 0 to $N/2 - 1$ spanned by β_q . Therefore,

$$\tilde{\rho}_{00}^{(n-1)}(r) \approx \frac{1}{r} \times \frac{r}{2} = 0.5, \quad r \text{ even}. \quad (20)$$

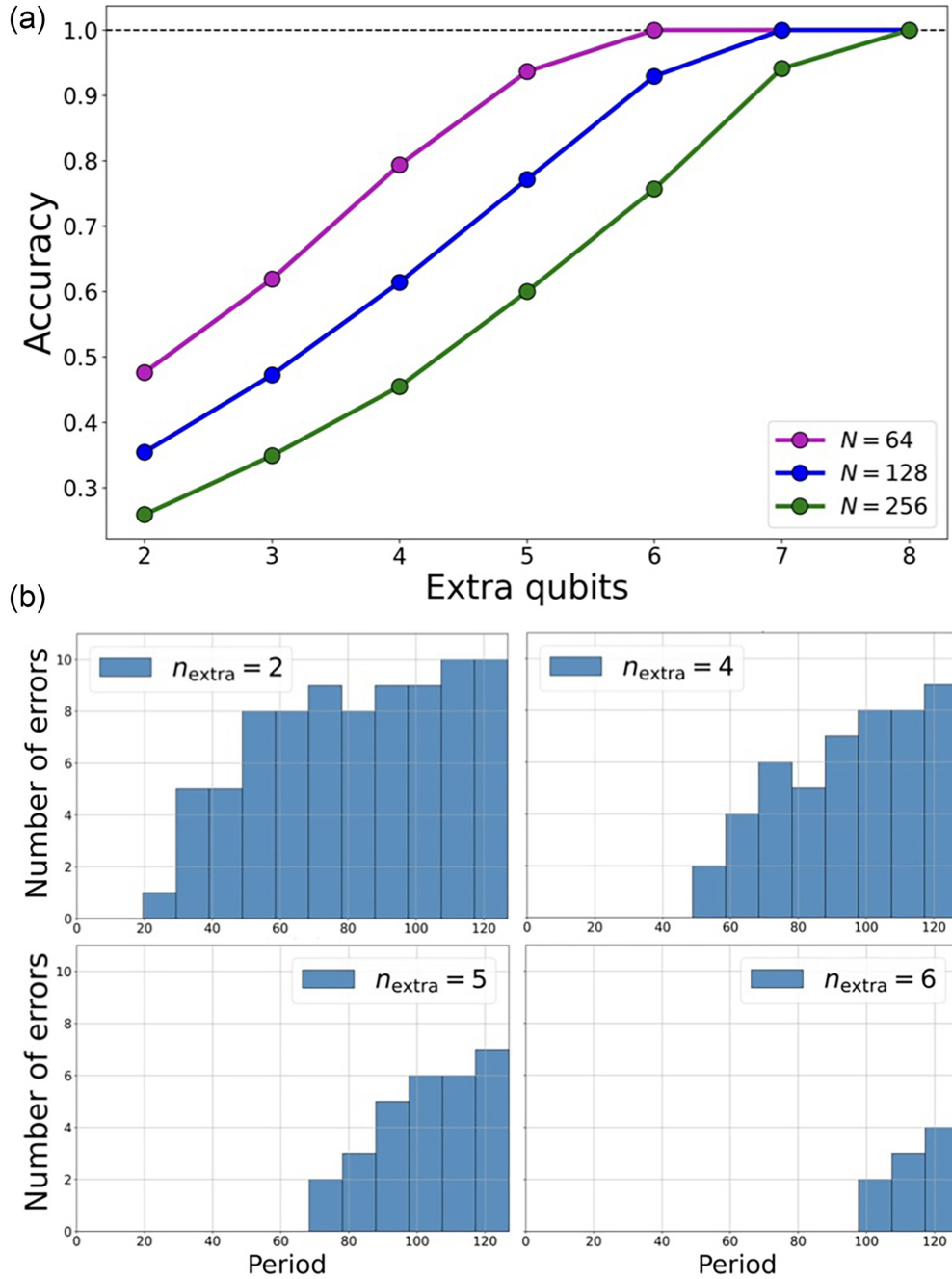


FIG. 4. (a) Accuracy of quantum period finding from 1-RDMs using our numerical approach. Results are shown for $N=64$, 128, 256, which corresponds to $n=6$, 7, 8 qubits, respectively, by plotting the accuracy as a function of the number of extra qubits added to improve the root-finding procedure. (b) Histograms of number of errors vs position of the period in the domain, shown for the case of $n=7$ ($N=128$) for four different numbers of extra qubits (n_{extra}).

Together with Eq. (19), this explains the pattern observed in the last qubit, where $a_z > 0$ only for odd periods, while $a_z = 0$ for even periods.

Next, we focus on the penultimate qubit, $q = n - 2$. For that qubit, we do not examine odd periods as we can already find those using the last qubit. Rather, we focus on the new peaks $a_z^{(n-2)}(r) > 0$ that are not present in the last qubit. These peaks occur for even periods, $r = 2r'$, with odd $r' < N/2$, for which the approximation in Eq. (17) holds [see Fig. 3(b)]. For these periods, we can write $jN/r = j(N/2)/r'$ with r' odd and repeat the reasoning above but now on half of the domain, with size

$N/2$. For $q = n - 2$, β_q is the set of strings with digit 0 in the penultimate qubit. Therefore, $\beta_q = \{(b_{n-1} 0 b_{n-3} \dots b_0)\}$ spans the first and third quarters of the domain, as shown schematically in Fig. 5(b). (For example, for four qubits, $N = 2^n = 16$, and $\beta_{q=n-2} = \{0, 1, 2, 3, 8, 9, 10, 11\}$.) Consider the values $j(N/2)/r'$, with $j = 0, 1, \dots, 2r' - 1$. Since r' is odd, more than half of these values will fall in the first and third quarters of the domain, which are spanned by $\mathcal{R}(\beta_q)$.

For example, consider the case $r = 2r' = 14$ ($r' = 7$) in Fig. 5(b). The values $j(N/2)/r'$ in the first quarter of the domain, $[0, N/4)$, are $(N/2) \times \{0, 1/7, 2/7, 3/7\}$,

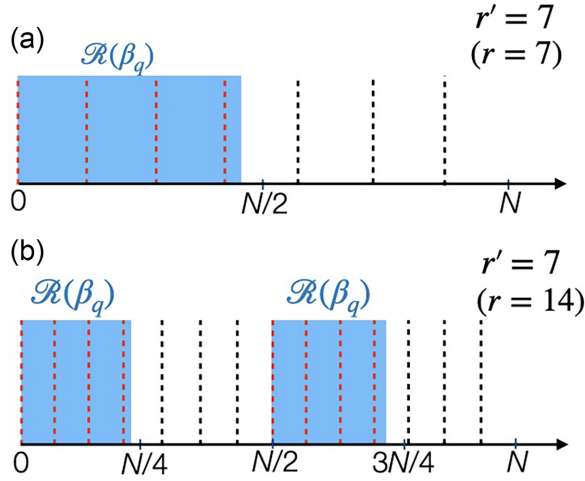


FIG. 5. Relation between the range $\mathcal{R}(\beta_q)$ and the values jN/r for (a) the last qubit $q=n-1$, for $r=7$, and (b) the next to last qubit, $q=n-2$, for $r=2r'$, with $r'=7$. The range of real values spanned by the strings in β_q , $\mathcal{R}(\beta_q)$, is a set of continuous intervals, shown using blue color. The vertical dashed lines represent jN/r , for $j=0, 1, \dots, r-1$, with the values overlapping with $\mathcal{R}(\beta_q)$ highlighted in red.

and those in the third quarter, $[N/2, 3N/4]$, are $(N/2) \times \{1, 8/7, 9/7, 10/7\}$. In both quarters, we have $\lceil r'/2 \rceil = 4$ values of jN/r inside $\mathcal{R}(\beta_q)$. Adding these contributions, from Eq. (16) we get

$$\tilde{\rho}_{00}^{(n-2)}(r) \approx \frac{2}{r} \lceil (r/2)/2 \rceil > 0.5, \quad r = 2r', \text{ odd } r' < N/2. \quad (21)$$

This is the same result as Eq. (19), as can be seen by substituting $r = 2r'$, and it agrees with the approximate formula for the 1-RDM diagonal elements [Eq. (17)].

Finally, we generalize this formula to all qubits. In our algorithm, we use the qubit q' from the last, $q=n-1-q'$, to find the periods $r=2^{q'}r'$, with r' odd and smaller than $N/2^{q'}$. For that qubit, we have $jN/r = j(N/2^{q'})/r'$ and the integers in β_q span $2^{q'}$ intervals of size $N/2^{q'+1}$, with the initial points of adjacent intervals spaced apart by $N/2^{q'}$. Because r' is odd and smaller than $N/2^{q'}$, there are $\lceil r'/2 \rceil = \lceil (r/2^{q'})/2 \rceil$ values of jN/r in each of the intervals spanned by $\mathcal{R}(\beta_q)$, for a total of $2^{q'} \lceil (r/2^{q'})/2 \rceil$ values of jN/r . Therefore, using Eq. (16), we obtain

$$\tilde{\rho}_{00}^{(n-1-q')}(r) \approx \frac{2^{q'}}{r} \lceil (r/2^{q'})/2 \rceil, \quad (22)$$

which is exactly Eq. (17), the approximate formula for the 1-RDM diagonal elements we set out to derive. In our algorithm, we use this formula to approximate the peaks of $\rho_{00}^{(n-1-q')}(r)$ at periods $r=2^{q'}r'$, with odd $r' < N/2^{q'}$. This allows us to find arbitrary periods in the domain.

D. Effects of noise on period-finding accuracy

While our period-finding approach based on 1-RDMs is developed primarily as a computational tool to use in combination with approximate classical simulations, 1-RDMs measured in real devices will be affected by noise and

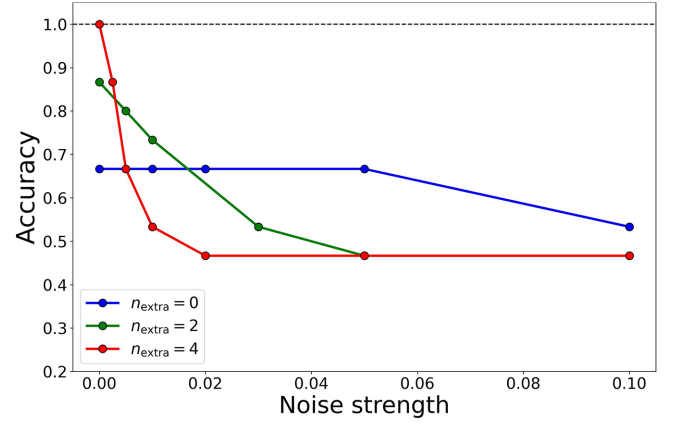


FIG. 6. Period-finding accuracy in the presence of depolarizing noise, shown for the case of $n=4$ qubits, for noise strengths up to 0.1 and for three different values of extra qubits, $n_{\text{extra}} = 0, 2$, and 4.

finite sampling. Therefore, we explore the effects of noise on the accuracy of the proposed period-finding method. In the presence of noise or finite-sampling errors, the peak structure of the 1-RDMs is expected to deviate from the exact one, reducing the period-finding accuracy. To illustrate this point, we simulate the 1-RDMs in the presence of depolarizing noise and analyze the accuracy of period finding as a function of noise strength. To achieve this, we add depolarizing noise to the one-qubit Hadamard gates in the QFT; for our state-vector simulations, this is implemented by adding a one-qubit gate, randomly chosen in set of Pauli X , Y , and Z , with probability p after each Hadamard gate, where p is the noise strength [10].

We find that the peak structure of the 1-RDMs remains unchanged in the presence of noise, but the values of the peaks shift slightly depending on the noise strength. To quantify the impact of noise, we analyze the period-finding accuracy as a function of depolarizing noise strength using 1-RDMs obtained for different values of noise strength between 0% and 10% (p from 0 to 0.1). These results are shown in Fig. 6 for $n=4$ qubits for three values of extra qubits between 0 and 4. The case with no extra qubits has an accuracy of 67% in the absence of noise, which is unchanged up to large noise values of 5% (noise strength $p=0.05$).

When extra qubits are added to improve the accuracy, in the absence of noise, the approach reaches 100% accuracy for $2n$ qubits (here, four extra qubits), but the effect of noise is stronger in the presence of extra qubits because the QFT circuit becomes deeper. For example, for two extra qubits, the accuracy without noise is 87%, versus 73% for a noise level of 1%. For four extra qubits, the approach reaches 100% accuracy without noise, but the accuracy decreases to 53% for 1% noise. This result shows that in the presence of noise, there is a trade-off between adding extra qubits to improve the accuracy and lowering the qubit count to reduce noise effects.

IV. DISCUSSION

Our treatment of QPF focuses on finding the period of a function using only single-qubit quantities, the diagonal elements of the 1-RDMs for all qubits in the first register of the QPF circuit. These one-qubit marginals, $\rho_{00}^{(q)}$,

correspond to the probability of measuring 0 for qubit q when conducting single-qubit measurements and consist of a set of n real numbers between 0 and 1. We have shown that these single-qubit quantities suffice to find the period of a generic periodic function.

This approach is different from the standard QPF algorithm, which samples bit strings in the output of the QPF circuit and uses reduced fraction analysis to obtain the period. In both approaches, for a function defined in the domain of n -bit integers, one needs $\sim 2n$ qubits in the first register to obtain the period reliably. Our approach is also distinct from both partial measurement [33] and tomography-based methods [34–36]. In particular, our approach does not rely on partial measurements but instead exploits the structure of the 1-RDMs in the output of the QFT, and it uses only single-qubit quantities, the 1-RDMs, thereby avoiding both global measurements and the overhead of full state reconstruction.

We briefly discuss two implications of these results. First, similarly to other quantum algorithms, the output state of the QPF circuit is concentrated around a set of high-probability bit strings. (In the case of QPF, the output is dominated by r bit strings, where the period r can take any value from 1 to $2^n - 1$.) However, our results show that thinking in terms of the entire Hilbert space of bit strings is not necessary because the 1-RDM one-qubit marginals contain the same information about the period as the entire set of bit strings. Since the 1-RDM diagonal elements for n qubits are only n numbers, as opposed to $r \approx \mathcal{O}(2^n)$ bit strings, conceptually our approach can be viewed as a “compression” of the information in the QPF algorithm.

Second, our approach works for generic periodic functions, including the modular exponential function used in Shor’s quantum algorithm for integer factorization. Since calculating *exact* 1-RDMs from state-vector simulations requires exponential resources that scale as $\mathcal{O}(2^n)$, there is no benefit in extracting the period from exact 1-RDMs as opposed to bit strings sampled from state-vector simulations. However, *approximate* simulations able to obtain the 1-RDMs with high

accuracy may become available in the future, for example, using methods based on tensor networks or open quantum systems. Combined with the approach described in this work, hypothetical classical simulations—not yet available to our knowledge—capable of predicting accurate 1-RDM marginals in polynomial time for the QPF circuit may enable more efficient classical algorithms for period finding and integer factorization. Preliminary results, to be reported elsewhere, suggest that the simple scheme for approximate 1-RDM simulations proposed in Ref. [30] may suffice to predict a subset of peaks in the 1-RDMs and find the corresponding periods.

In summary, we show an approach to find the period of a function in the domain of n -bit integers, a quantity with formal exponential complexity of $\mathcal{O}(2^n)$, using only n one-qubit marginals. This is achieved by finding patterns in the 1-RDMs, formulating reliable approximations, and developing a numerical approach to obtain the period from the 1-RDMs. Future work will focus on the development of accurate approximations for one- and multiqubit RDMs for QPF and other quantum algorithms.

ACKNOWLEDGMENTS

M.B. gratefully acknowledges Vittorio Giovannetti for guidance and fruitful discussions. M.B. thanks the Scuola Normale Superiore in Pisa, Italy, for hosting him during a sabbatical that led to the development of this project. This work was partially supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of Basic Energy Sciences, Scientific Discovery through Advanced Computing (SciDAC) program under Award No. DE-SC0022088 for method development.

DATA AVAILABILITY

The data that support the findings of this article are openly available [31].

-
- [1] P. W. Shor, Algorithms for quantum computation: Discrete logarithms and factoring, in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science* (IEEE, Los Alamitos, CA, USA, 1994), pp. 124–134.
 - [2] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J. Comput.* **26**, 1484 (1997).
 - [3] A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller, Practical quantum advantage in quantum simulation, *Nature (London)* **607**, 667 (2022).
 - [4] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush, N. Ding, Z. Jiang, M. J. Bremner, J. M. Martinis, and H. Neven, Characterizing quantum supremacy in near-term devices, *Nat. Phys.* **14**, 595 (2018).
 - [5] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, Quantum supremacy using a programmable superconducting processor, *Nature (London)* **574**, 505 (2019).
 - [6] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, *et al.*, Quantum computational advantage using photons, *Science* **370**, 1460 (2020).
 - [7] J. I. Cirac and P. Zoller, Goals and opportunities in quantum simulation, *Nat. Phys.* **8**, 264 (2012).
 - [8] I. M. Georgescu, S. Ashhab, and F. Nori, Quantum simulation, *Rev. Mod. Phys.* **86**, 153 (2014).
 - [9] J. Preskill, Quantum computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
 - [10] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th anniversary ed. (Cambridge University Press, New York, 2010).
 - [11] S. Barnett, *Quantum Information* (Oxford University Press, New York, 2009).
 - [12] For generic periods, testing if $f(x) = f(x + r)$ requires computing the function outside the domain, for values of x up to $2N$ if the function is known to be periodic.

- [13] H. De Raedt, F. Jin, D. Willsch, M. Willsch, N. Yoshioka, N. Ito, S. Yuan, and K. Michielsen, Massively parallel quantum computer simulator, eleven years later, *Comput. Phys. Commun.* **237**, 47 (2019).
- [14] M. Smelyanskiy, N. P. D. Sawaya, and A. Aspuru-Guzik, qHIPSTER: The quantum high performance software testing environment, *arXiv:1601.07195*.
- [15] T. Jones, A. Brown, I. Bush, and S. C. Benjamin, QuEST and high performance simulation of quantum computers, *Sci. Rep.* **9**, 10736 (2019).
- [16] Z.-Y. Chen, Q. Zhou, C. Xue, X. Yang, G.-C. Guo, and G.-P. Guo, 64-qubit quantum circuit simulation, *Sci. Bull.* **63**, 964 (2018).
- [17] I. L. Markov and Y. Shi, Simulating quantum computation by contracting tensor networks, *SIAM J. Comput.* **38**, 963 (2008).
- [18] N. Yoran and A. J. Short, Efficient classical simulation of the approximate quantum Fourier transform, *Phys. Rev. A* **76**, 042321 (2007).
- [19] Y. Zhou, E. M. Stoudenmire, and X. Waintal, What limits the simulation of quantum computers? *Phys. Rev. X* **10**, 041038 (2020).
- [20] Y.-T. Chen, C. Farquhar, and R. M. Parrish, Low-rank density-matrix evolution for noisy quantum circuits, *npj Quantum Inf.* **7**, 61 (2021).
- [21] A. Li, O. Subasi, X. Yang, and S. Krishnamoorthy, Density matrix quantum circuit simulation via the BSP machine on modern GPU clusters, in *Sc20: International Conference for High Performance Computing, Networking, Storage and Analysis* (IEEE, Piscataway, NJ, USA, 2020), pp. 1–15.
- [22] B. Jónsson, B. Bauer, and G. Carleo, Neural-network states for the classical simulation of quantum computing, *arXiv:1808.05232*.
- [23] X. Gao and L.-M. Duan, Efficient representation of quantum many-body states with deep neural networks, *Nat. Commun.* **8**, 662 (2017).
- [24] S. Bravyi and D. Gosset, Improved classical simulation of quantum circuits dominated by Clifford gates, *Phys. Rev. Lett.* **116**, 250501 (2016).
- [25] G. Vidal, Efficient classical simulation of slightly entangled quantum computations, *Phys. Rev. Lett.* **91**, 147902 (2003).
- [26] T. Ayril, T. Louvet, Y. Zhou, C. Lambert, E. M. Stoudenmire, and X. Waintal, Density-matrix renormalization group algorithm for simulating quantum circuits with a finite fidelity, *PRX Quantum* **4**, 020304 (2023).
- [27] G. Lindblad, On the generators of quantum dynamical semi-groups, *Commun. Math. Phys.* **48**, 119 (1976).
- [28] H.-P. Breuer and F. Petruccione, *The Theory of Open Quantum Systems* (Oxford University Press, New York, 2002).
- [29] D. Manzano, A short introduction to the Lindblad master equation, *AIP Adv.* **10**, 025106 (2020).
- [30] M. Bernardi, Efficient mean-field simulations of quantum circuits inspired by density functional theory, *J. Chem. Theory Comput.* **19**, 8066 (2023).
- [31] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/bcxp-1qlld> for numerical routines and datasets generated in this work, provided as a Python Jupyter notebook.
- [32] M. T. Heath, *Scientific Computing: An Introductory Survey*, 2nd ed. (SIAM, New York, 2018).
- [33] R. B. Griffiths and C.-S. Niu, Semiclassical Fourier transform for quantum computation, *Phys. Rev. Lett.* **76**, 3228 (1996).
- [34] K. Banaszek, G. M. D'Ariano, M. G. A. Paris, and M. F. Sacchi, Maximum-likelihood estimation of the density matrix, *Phys. Rev. A* **61**, 010304(R) (1999).
- [35] M. Cramer, M. B. Plenio, S. T. Flammia, R. Somma, D. Gross, S. D. Bartlett, O. Landon-Cardinal, D. Poulin, and Y.-K. Liu, Efficient quantum state tomography, *Nat. Commun.* **1**, 149 (2010).
- [36] H.-Y. Huang, R. Kueng, and J. Preskill, Predicting many properties of a quantum system from very few measurements, *Nat. Phys.* **16**, 1050 (2020).