


## Using quantum computers to identify prime numbers via entanglement dynamics

Victor F. dos Santos<sup>✉\*</sup> and Jonas Maziero<sup>✉†</sup>

*Physics Department, Center for Natural and Exact Sciences, Federal University of Santa Maria, Santa Maria, RS 97105-900, Brazil*

 (Received 22 March 2024; accepted 9 July 2024; published 2 August 2024)

Recently, the entanglement dynamics of two harmonic oscillators initially prepared in a separable-coherent state was demonstrated to offer a pathway for prime number identification. This article presents a generalized approach and outlines a deterministic algorithm making possible the implementation of this theoretical concept on scalable fault-tolerant qubit-based quantum computers. We prove that the diagonal unitary operations employed in our algorithm exhibit a polynomial-time complexity of degree two, contrasting with the previously reported exponential complexity of general diagonal unitaries.

DOI: [10.1103/PhysRevA.110.022405](https://doi.org/10.1103/PhysRevA.110.022405)

### I. INTRODUCTION

The quest to reliably and efficiently identify prime numbers (PNs) remains a topic of great interest in number theory [1–4], particularly due to its intriguing connection with the nontrivial zeros of Riemann’s zeta function [5–8]. Over the centuries, numerous classical algorithms have been devised for identifying primes, each offering its own set of advantages and limitations [1,2,9]. Among these, the AKS primality test stands out as the first deterministic algorithm to exhibit polynomial-time complexity for verifying the primality of individual integers, albeit with a polynomial degree that renders it less efficient for larger numbers [10]. Conversely, the Sieve of Eratosthenes offers a simpler approach, focusing on identifying all PNs within a specified range  $N$ . Its time complexity,  $O(N \log \log N)$ , renders it particularly efficient for this purpose [2].

While classical algorithms for PNs identification [10–13] have undergone significant development, their adaptation to the realm of quantum computers (QCs) remains relatively limited [14–16]. However, the intersection of such questions with experimental physics [17–19] presents a promising avenue for the development of more intuitive quantum algorithms. A notable recent study [19] proposed an innovative approach to primality testing using quantum optics. In their work, researchers devised an experiment involving the entanglement of two quantum harmonic oscillators initially prepared in coherent states, followed by the measurement of the reduced linear entropy of one of them. They theorized that information regarding PNs could be extracted from the Fourier modes of the reduced linear entropy: PNs were expected to adhere to a lower bound curve, while composite numbers would consistently surpass this bound. Although the experimental implementation has yet to be realized, and has known scalability limitations, their theoretical groundwork has laid the foundation for us to generalize their approach and to develop a

deterministic algorithm tailored for implementation on qubit-based QCs.

In this article, we build upon the theoretical framework proposed in Ref. [19], aiming to adapt it for implementation on qubit-based QCs by removing certain restrictions imposed on the Hamiltonian and initial states. As a result, we demonstrate, as detailed in the Appendixes, that the class of diagonal unitary gates utilized in our approach can be implemented in polynomial time, contrary to the expectations set forth in Refs. [20,21]. Our algorithm is designed to determine all PNs within a given range  $N$  through the manipulation of a bipartite system  $AB$  and the measurement of the linear entropy of entanglement [22–25] of subsystem  $A$  over a period  $T$ .

Our methodology unfolds with the following steps. First, we modify the definitions to align with the peculiarities of qubit-based QCs. Secondly, we select a suitable initial state that can be efficiently prepared. Thirdly, we efficiently prepare an evolved state using the techniques outlined in Ref. [21], which surprisingly results in exponential gate cost reduction in comparison to the general case. Subsequently, we measure the reduced purity, a task that can be executed efficiently [26]. Following this, we calculate the Fourier modes of the reduced purity function via numerical integration methods [27].

Given a data set encompassing all points within the range  $N$ , our algorithm enables the deterministic identification of Fourier modes corresponding to PNs, allowing for the distinction between primes and composites. We quantify the number of gates utilized at each step, with a specific focus on  $Z$  rotations, controlled-NOT, and Hadamard gates. Additionally, we discuss simulations conducted using QISKIT [28] and explore potential enhancements to our algorithm for more efficient implementation on real qubit-based QCs.

We begin by establishing key definitions. Let  $A$  and  $B$  represent the respective subsystems, each characterized by a time-independent Hamiltonian  $\hat{H}_A$  and  $\hat{H}_B$ , where  $\hat{H}_A = \hat{H}_B$ . We define a bipartite Hamiltonian  $\hat{H}_{AB} = \lambda \hat{H}_A \otimes \hat{H}_B$ , with  $\lambda \in \mathbb{R}$  denoting the coupling constant. The corresponding time-evolution operator is given by  $\hat{U}(t) = e^{-i\hat{H}_{AB}t/\hbar}$  [29].

\*Contact author: [victorfds997@gmail.com](mailto:victorfds997@gmail.com)

†Contact author: [jonas.maziero@ufsm.br](mailto:jonas.maziero@ufsm.br)

To obtain a distinction between prime and composite numbers, we employ the initial state  $|\psi(0)\rangle_{AB} = |\phi\rangle_A \otimes |\phi\rangle_B$ . A suitable choice for these individual states is

$$|\phi\rangle_S = \sum_{n_S=1}^d c_{n_S} |E_{n_S}\rangle, \quad (1)$$

where  $S = A, B$  represents the subsystem index,  $d$  is the dimension of each subsystem,  $c_{n_S} \neq 0$  are the initial-state coefficients, and  $\{|E_{n_S}\rangle\}_{n_S=1}^d$  and  $\{|E_{n_S}\rangle\}_{n_S=1}^d$  denote the eigenvalues and eigenvectors of each subsystem Hamiltonian, respectively. The evolved state at time  $t$  is  $|\psi(t)\rangle_{AB} = \hat{U}(t)|\psi(0)\rangle_{AB} = \sum_{n_A, n_B=1}^d c_{n_A} c_{n_B} e^{-i\lambda E_{n_A} E_{n_B} t/\hbar} |E_{n_A} E_{n_B}\rangle$ .

Our main condition requires that the energy levels of both individual Hamiltonians are equidistant, i.e.,  $E_{n_S} = n_S \mu$  for some constant  $\mu \in \mathbb{R}$ . Defining  $\omega = \lambda \mu^2 / \hbar$ , we find that

$$\hat{U}(t) = \sum_{n_A, n_B=1}^d e^{-i\omega n_A n_B t} |E_{n_A} E_{n_B}\rangle \langle E_{n_A} E_{n_B}| \quad (2)$$

and

$$|\psi(t)\rangle_{AB} = \sum_{n_A, n_B=1}^d c_{n_A} c_{n_B} e^{-i\omega n_A n_B t} |E_{n_A} E_{n_B}\rangle. \quad (3)$$

A key result of our research is the demonstration of high gate efficiency for implementing the diagonal unitary gate specified in Eq. (2), as detailed in Appendix D. We show that the gate cost for constructing the  $q$ -qubit unitary gate  $\hat{U}(t)$  using this method is a polynomial function  $G_2(q) = \frac{3}{4}q^2 + q$ . This result not only facilitates PNs identification but also paves the way for efficient implementation of similar unitary gates in future qubit-based QCs research.

The remainder of this article is organized as follows. In Sec. II, we give the general expression for the reduced purity of a subsystem (A) and highlight its mathematical properties. Next, in Sec. III, we establish the theoretical connection between the Fourier modes of the reduced purity and the distribution of prime numbers. Then, in Sec. IV, we report our quantum algorithm, specifying the techniques used and the associated computational costs. In Sec. V, we present the results of simulations made using QISKIT. Finally, we conclude in Sec. VI by revisiting the key points of our method and quantum algorithm, while discussing limitations and further potential improvements with respect to an implementation on quantum hardware. Additional details supporting our findings are provided in the Appendices. Appendix A is a summary of the technique developed in Ref. [21] for the implementation of general diagonal unitary gates using Walsh functions. Appendix B presents a proof for a known identity that relates tensor products of Pauli  $\hat{Z}$  gates and  $\overline{\text{CNOT}}$ s, and a proof for how this identity relates to the implementation of exponentials of Walsh operators. In Appendix C, we prove some results regarding Walsh matrices and delineate our notation for them, as it will be heavily used in further demonstrations. Then, Appendix D uses results from the previous Appendices to rigorously demonstrate that the diagonal unitary gate in Eq. (2) may be implemented efficiently using only a polynomial number (with respect to the number of qubits) of elementary gates. Furthermore, Appendix E is a direct proof for a modified

version of the SWAP test, aiming for the estimation of the reduced purity.

## II. REDUCED PURITY

Without loss of generality, we designate subsystem  $A$  for computing the reduced purity  $\gamma_A(t)$ . Let us begin by revisiting the definition of the reduced density operator  $\hat{\rho}_A(t)$  for a system  $AB$  with density operator  $\hat{\rho}_{AB}(t)$ , given as  $\hat{\rho}_A(t) = \text{Tr}_B(\hat{\rho}_{AB}(t))$ , where  $\text{Tr}_B(\cdot)$  denotes the partial trace function [30] over subsystem  $B$ . The reduced purity function,  $\gamma_A(t) = \text{Tr}(\hat{\rho}_A^2(t))$ , can then be computed from  $\hat{\rho}_A(t)$ . This quantity is related to the linear entanglement entropy by  $E_l(|\psi(t)\rangle_{AB}) = 1 - \gamma_A(t)$ . Given that  $\hat{\rho}_{AB}(t) = |\psi(t)\rangle_{AB} \langle \psi(t)|$ , the reduced purity can be straightforwardly expressed as

$$\gamma_A(t) = \sum_{j,k,l,m=1}^d |c_j|^2 |c_k|^2 |c_l|^2 |c_m|^2 e^{-i\omega t(j-k)(l-m)}. \quad (4)$$

It is noteworthy to highlight several properties of the function  $\gamma_A(t)$  defined in Eq. (4). First, it exhibits time periodicity with period  $T = 2\pi/\omega$ , a characteristic stemming directly from the time evolution of our system. Secondly, a notable observation arises from the structure of the sum in Eq. (4): the indices  $j, k, l$ , and  $m$  all take the same values. As a consequence, the imaginary parts of the phases  $e^{-i\omega t(j-k)(l-m)}$  for a fixed  $t$  mutually cancel each other. This cancellation is crucial, ensuring that  $\gamma_A(t)$  remains a real-valued function. This function is symmetric about half the period,  $\gamma_A(T/2 + h) = \gamma_A(T/2 - h)$ , which enables us to halve the number of times we need to execute the quantum circuit to obtain it.

## III. MAPPING PRIME NUMBERS WITH FOURIER MODES

The reduced purity function given by Eq. (4) can be expressed as a finite sum of cosines, where the maximum number of Fourier modes  $\alpha_n$  is  $(d-1)^2$ . Therefore, employing a Fourier expansion in this scenario yields

$$\gamma_A(t) = \alpha_0 + \sum_{n=1}^{(d-1)^2} \alpha_n \cos(n\omega t), \quad (5)$$

where  $\alpha_0$  represents the average value and  $\alpha_n$  are the Fourier modes [31].

To compute the Fourier modes  $\alpha_n$ , we utilize the expression

$$\alpha_n = 4 \sum_{k,m=1}^{d-1} \sum_{j>k}^d \sum_{l>m}^d |c_j|^2 |c_k|^2 |c_l|^2 |c_m|^2 \delta_{(j-k)(l-m)}^n, \quad (6)$$

where  $\delta_{(j-k)(l-m)}^n$  represents the Kronecker delta function ensuring the resonance condition for the Fourier modes. This formulation allows us to decompose the reduced purity  $\gamma_A(t)$  into its constituent Fourier components, facilitating the identification of PNs based on their distinct Fourier signatures.

For prime  $n$ , the trivial decomposition is  $(j-k) = n$  and  $(l-m) = 1$ , and vice versa. This results in a unique decomposition that corresponds to the expected behavior for prime numbers. However, if  $n$  is composite, it possesses nontrivial decompositions as well. To examine the impact of these decompositions on the Fourier modes expressed in Eq. (6), let us

define the lower bound  $B_n$  as the value obtained from Eq. (6) using the trivial decomposition of  $n \geq 2$ . Hence we have

$$B_n = 8 \sum_{k=1}^{d-n} \sum_{m=1}^{d-1} |c_k|^2 |c_m|^2 |c_{k+n}|^2 |c_{m+1}|^2. \quad (7)$$

This lower bound  $B_n$  provides insight into the minimum value that the Fourier coefficient  $\alpha_n$  can attain for a given composite  $n$ . Understanding this bound is crucial for discerning the distinct Fourier signatures associated with prime and composite numbers. For  $2 \leq n \leq d-1$ , we have  $B_n > 0$  as per Eq. (7). However, when  $d \leq n \leq (d-1)^2$ , the domain of  $B_n$  can be extended such that  $B_n = 0$ .

Now, let  $\{y_r^{(n)}\}_{r=1}^z$  represent the sequence of  $z$  distinct divisors of  $n \geq 2$  in increasing order of magnitude. Excluding the trivial cases  $y_1^{(n)} = 1$  and  $y_z^{(n)} = n$ , we find that in general

$$\alpha_n = B_n + 4 \sum_{r=2}^{z-1} \sum_{k=1}^{d-\frac{n}{y_r^{(n)}}} \sum_{m=1}^{d-y_r^{(n)}} |c_k|^2 |c_m|^2 |c_{k+\frac{n}{y_r^{(n)}}}|^2 |c_{m+y_r^{(n)}}|^2. \quad (8)$$

This expression for  $\alpha_n$  encompasses both the contribution from the trivial decomposition and the contributions from the nontrivial divisors of  $n$ , enabling a comprehensive assessment of the Fourier modes associated with composite numbers.

In the domain  $2 \leq n \leq 2(d-1)$ , we can confidently assert that  $\alpha_n > B_n$  holds true. However, beyond this range, specifically in the interval  $2(d-1) < n \leq (d-1)^2$ , certain composite numbers  $n = n_0$  may exhibit  $\alpha_{n_0} = 0$ . This phenomenon arises because the first semiprimes (numbers that are the product of two prime numbers) are multiples of 2. Consequently, for  $n_0 = 2v$ , where  $v > d-1$  is a prime, there exist no values for the indices  $k$  and  $m$  in Eq. (8) that fall within their defined ranges in the summation. However, in this interval it is possible to discard any integer as a prime candidate if it has a nonzero Fourier mode. Since prime numbers always yield  $\alpha_n = 0$  in this interval, we can safely guarantee that if  $\alpha_n \neq 0$ , then  $n$  is composite. The inverse, however, is not always true: some composite numbers have  $\alpha_n = 0$ .

Here is the summary of the expected values of  $\alpha_n$  in the three regimes.

**Regime I:**  $2 \leq n \leq d-1$ . For prime numbers in this range, it holds true that  $\alpha_n = B_n > 0$ ; otherwise,  $\alpha_n > B_n$ .

**Regime II:**  $d \leq n \leq 2(d-1)$ . Prime numbers in this interval exhibit  $\alpha_n = B_n = 0$ , while composite numbers consistently demonstrate  $\alpha_n > 0$ .

**Regime III:**  $2(d-1) < n \leq (d-1)^2$ . Prime numbers within this regime always yield  $\alpha_n = 0$ . However, some composite numbers may also yield  $\alpha_n = 0$  in this interval. Consequently, this regime cannot provide conclusive evidence regarding the primality of  $n$ . Nonetheless, any integer  $n$  with  $\alpha_n \neq 0$  in this regime is guaranteed to be a composite number.

This summary provides a clear delineation of the behavior of  $\alpha_n$  across different regimes, aiding in the identification of prime numbers based on their Fourier modes.

Our regime of interest is  $\mathcal{D} = \mathbf{I} \cup \mathbf{II}$ . In  $\mathcal{D}$ , it is consistently true that

$$\alpha_n \geq B_n, \quad (9)$$

with equality achieved if and only if  $n$  is a prime number. This inequality forms the cornerstone of the algorithm and serves

as the basis for objectively distinguishing prime numbers from composites.

While the protocol enables the computation of  $\alpha_n$ , without knowledge of  $B_n$  in Regime **I**, it is impossible to discern whether  $\alpha_n = B_n$  or  $\alpha_n \neq B_n$ . A straightforward solution involves obtaining the analytical value of the lower bound  $B_n$  within that regime, achievable by selecting a simple initial state and utilizing Eq. (7) subsequently. In our algorithm, for simplicity, we opt for an initial state of maximum superposition.

#### IV. THE QUANTUM ALGORITHM

Below, we provide a structured description of all the steps necessary to develop our protocol. We also present here the number of gates necessary for each step.

**1. Qubit codification.** To adapt our protocol to a qubit-based quantum computing algorithm, we need to adjust some of our definitions regarding the translation of qudits to qubits. Given that the bipartite system  $AB$  has  $d^2$  energy levels and we aim to utilize  $q$  qubits instead of two qudits, the condition is imposed that

$$d^2 = 2^q. \quad (10)$$

Equation (10) inherently assumes that  $d$  is a power of 2. If  $d$  is not a power of 2, we have to find  $q$  such that  $q = 2 \lceil \log_2(d) \rceil$ , where  $\lceil \cdot \rceil$  denotes the ceiling function. We conveniently assign the first half of qubits to represent subsystem  $A$  and the remaining half to represent subsystem  $B$ .

**2. Initial state flexibility.** The initial state  $|\psi(0)\rangle_{AB}$  is defined as the product state  $|\psi(0)\rangle_{AB} = |\phi\rangle_A \otimes |\phi\rangle_B$ , where the coefficients  $c_{n_s}$  of the subsystem states  $|\phi\rangle_S = \sum_{n_s=1}^d c_{n_s} |E_{n_s}\rangle$  must satisfy  $c_{n_s} \neq 0$ . Leveraging this degree of freedom, we opt for convenience by employing an initial state that achieves maximum superposition, expressed as  $|\psi(0)\rangle_{AB} = \frac{1}{d} \sum_{n_A, n_B=1}^d |E_{n_A} E_{n_B}\rangle$ . Here, we implicitly define the eigenbasis  $\{E_{n_s}\}_{n_s=1}^d$  as the computational basis for each set of  $q/2$  qubits. To produce this initial state, we apply a series of Hadamard gates  $\hat{H}$  [29] to all  $q$  qubits:

$$|\psi(0)\rangle_{AB} = \hat{H}^{\otimes q} |000 \dots 0\rangle. \quad (11)$$

It is evident that the number of gates required here to generate this initial state is simply

$$G_1(q) = q. \quad (12)$$

**3. Evolved state preparation.** The detailed results regarding this item are provided in Appendix **A**. To obtain the evolved state  $|\psi(t)\rangle_{AB}$  of Eq. (3), we employ the method outlined in Ref. [21] to construct  $\hat{U}(t)$  efficiently. Initially, we have to determine, in principle, all the  $2^q - 1$  Walsh angles  $a_j(t)$  [32–35]. However, according to the results shown in Appendix **D**, only  $\frac{1}{4}q^2 + q$  of them are non-null. By definition, Walsh angles are expressed as

$$a_j(t) = \frac{1}{2^q} \sum_{k=0}^{2^q-1} f_k(t) w_{jk}, \quad (13)$$

where  $w_{jk}$  denotes the Paley-ordered discrete Walsh functions and  $f_k(t)$  are the eigenvalues of the operator  $\hat{f}(t)$ , extracted from  $\hat{U}(t) = e^{i\hat{f}(t)}$ .

Together with the Walsh angles  $a_j(t)$ , the unitary gate  $\hat{U}(t)$  is obtained using the formalism of Walsh operators  $\hat{w}_j$ . The expression for  $\hat{U}(t)$  is then given by

$$\hat{U}(t) = \prod_{j=1}^{2^q-1} e^{ia_j(t)\hat{w}_j}. \quad (14)$$

To produce the exponential operators  $e^{ia_j(t)\hat{w}_j}$ , we use the identity presented in Appendix B and consider the binary representation ( $j_q \dots j_2 j_1$ ) of the integer  $j$ , with the most significant nonzero bit (MSB) on the left. This enables us to represent the exponential operator as a single  $Z$  rotation,  $\hat{R}_z(\theta_j(t))$ , applied to qubit  $q_{m_j}$ , flanked by two identical controlled-NOT gates, with qubit  $q_{m_i}$  serving as the control and qubit  $q_{m_j}$  as the target. Here, the index  $m_{h_j} \geq 1$  signifies the position of the MSB of  $j$  and the indices  $m_i$  are defined by the condition  $j_{m_i} = 1$ . These  $\hat{R}_z(\theta_j(t))$  rotations have angles  $\theta_j(t) = -2a_j(t)$ .

Therefore, preparing  $|\psi(t)\rangle_{AB} = \hat{U}(t)|\psi(0)\rangle_{AB}$  demands a number of gates given by

$$G_2(q) = \frac{3}{4}q^2 + q. \quad (15)$$

**4. Reduced purity estimation.** This step involves efficiently obtaining the reduced purity  $\gamma_A(t)$  of Eq. (4) by utilizing techniques from Ref. [26]. The quantum circuit employed here resembles the SWAP test circuit [36,37] and employs an ancilla qubit  $q_0$  and two copies of  $q$  qubits prepared in the same pure state. The operations sequence for this quantum circuit is as follows: a Hadamard gate on  $q_0$ , qubit-qubit controlled-SWAP gates between the first  $q/2$  qubits of each copy, with  $q_0$  as the control qubit, another Hadamard gate on  $q_0$ , and a measurement of  $q_0$  in the computational basis. After repeatedly executing the circuit, we estimate the probability  $P_0$  of obtaining the state  $|0\rangle$  for  $q_0$ . Then, as detailed in Appendix E, the reduced purity over time can be estimated using the expression  $\gamma_A(t) = 2P_0(t) - 1$ .

This step involves a total number of gates given by

$$G_3(q) = \frac{3}{2}q + 2. \quad (16)$$

**5. Fourier modes calculation.** In Regime **I**, we obtain the lower bound  $B_n$  of Eq. (7) using the initial state of Eq. (11). In this case,  $c_j = \frac{1}{\sqrt{d}}$  for any  $j$  and the corresponding lower bound  $B_n$  interpolation in this range of  $n$  is a straight line with a negative slope. In Regime **II**, the lower bound is  $B_n = 0$ . The expression for  $B_n$  in the regime of interest  $\mathcal{D} = \mathbf{I} \cup \mathbf{II}$  can then be written as

$$B_n = \begin{cases} \frac{-8(d-1)}{d^4}n + \frac{8d-8}{d^3} & \text{if } n \in \mathbf{I}, \\ 0 & \text{if } n \in \mathbf{II}. \end{cases}$$

Considering the remarks made in the previous section, we know that in a graph of Fourier modes every prime number must have a corresponding  $\alpha_n$  position belonging exactly to the interpolated curve of  $B_n$ . Any composite number in the regime of interest  $\mathcal{D}$  has  $\alpha_n > B_n$  and thus is necessarily above  $B_n$ .

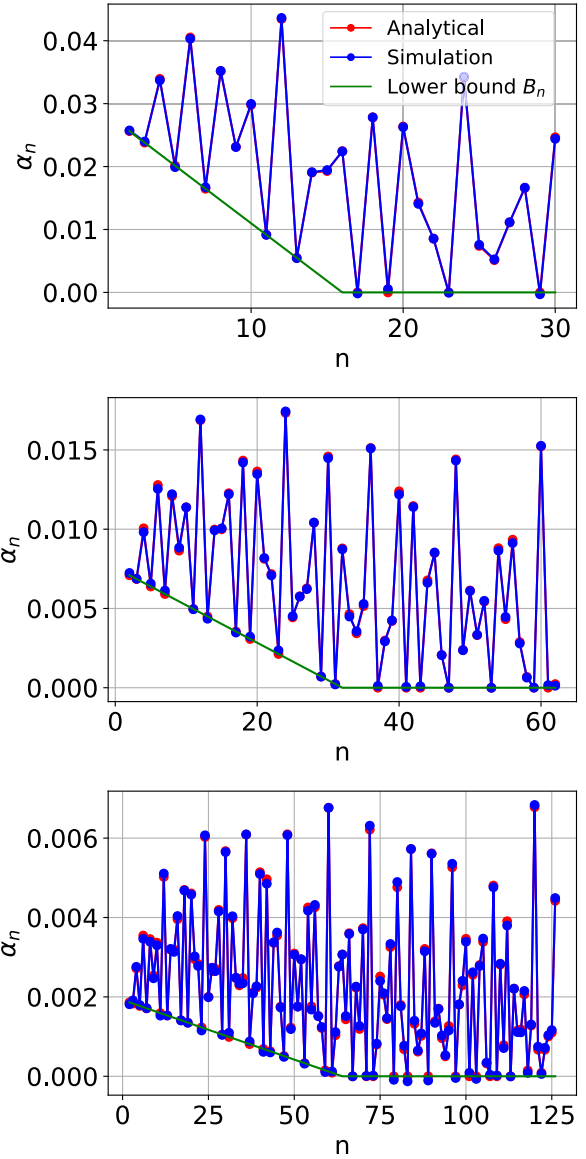


FIG. 1. Comparison of simulation results with theoretical predictions for the Fourier modes of the reduced purity across different dimensions ( $d = 16, 32, 64$ ). Red points represent the analytical values of  $\alpha_n$ , calculated directly from their theoretical expressions, while the green line stands for the minimum value for the Fourier modes, also derived from theoretical calculations. Blue points illustrate the Fourier modes obtained through numerical integration of the reduced purity  $\gamma_A(t)$  extracted from the classical emulation of our quantum circuit. The numerical integration was performed using various partition values ( $p = 375, 1500, 6000$ ). Prime numbers are expected to align with the lower bound  $B_n$ , whereas composite numbers appear above.

Now, using Fourier analysis, the Fourier modes  $\alpha_n$  are calculated by the integral

$$\alpha_n = \frac{2\omega}{\pi} \int_0^{T/2} \gamma_A(t) \cos(n\omega t) dt. \quad (17)$$

Normally, Eq. (17) would be an integral over the whole period  $T$ , but we are employing the property of the symmetry of  $\gamma_A(t)$ , presented earlier in this article. After calculating the

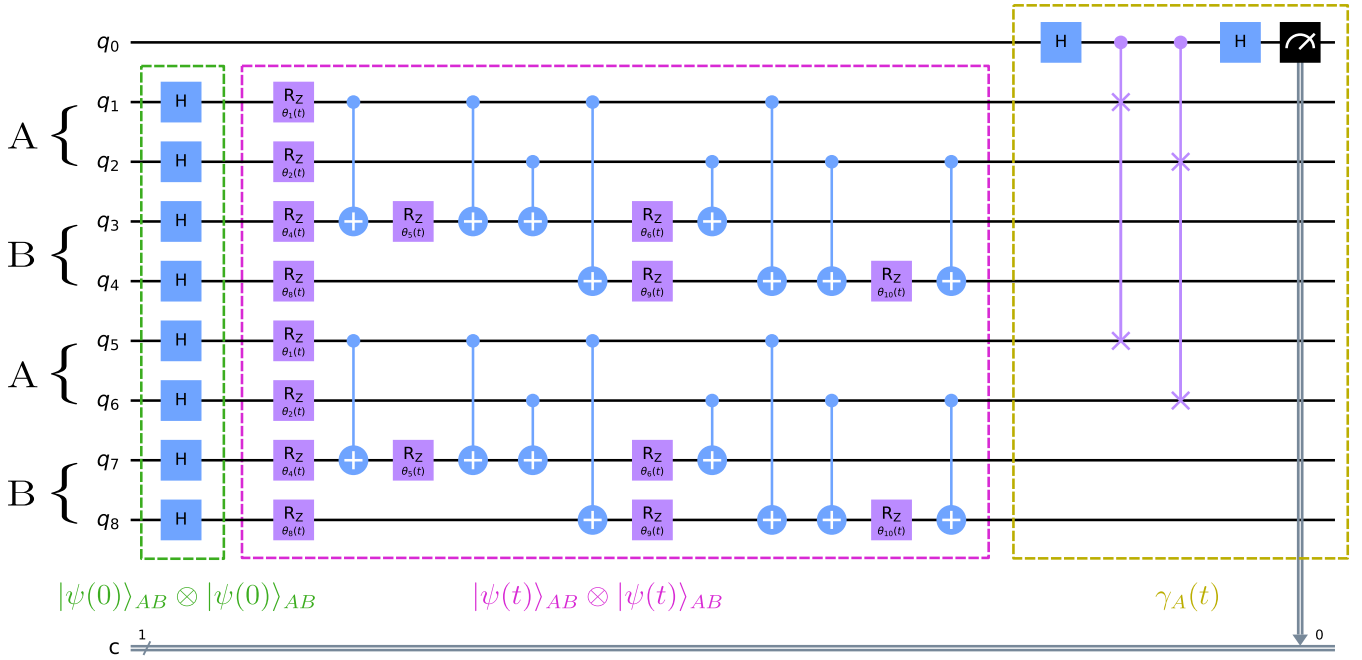


FIG. 2. Schematic of the quantum circuit for our algorithm for  $d = 4$ . Each colored box corresponds to a stage of the circuit and represents the set of operations in the respective step for the two copies of  $q$  qubits. In the first stage, we prepare the initial state of maximum superposition  $|\psi(0)\rangle_{AB}$  for both copies of  $q = 4$  qubits, starting from the state  $|0\rangle$  for each qubit. The second stage is used for the efficient state preparation of  $|\psi(t)\rangle_{AB}$  for each copy, where the total number of gates, including both copies, is given by  $\frac{3}{2}q^2 + 2q$ . In the last stage, with the aid of an ancilla qubit  $q_0$ , we apply the gates corresponding to the variation of the SWAP test to extract the reduced purity  $\gamma_A(t)$  after executing the circuit several times.

Fourier modes  $\alpha_n$ , the last part of our algorithm involves comparing the value of  $\alpha_n$  with the analytical lower bound  $B_n$ . In this final step, the numerical integration is done in  $p$  partitions, resulting in an equivalent number of points used for  $\gamma_A(t)$  in the interval  $0 \leq t \leq T/2$ . Consequently, to achieve a desired precision  $\epsilon$ , our quantum circuit requires at least  $p$  executions. Currently, the exact optimal scaling of  $p$  with respect to  $d$ , for a given  $\epsilon$ , remains undetermined.

### V. SIMULATIONS

In order to evaluate the applicability of our algorithm, classical simulations were performed using IBM’s QISKIT framework (version 0.45.1). These simulations targeted three distinct values of  $d$ , with results depicted in blue in Fig. 1. For all the simulations, we used  $10^5$  shots and fixed  $\omega = 0.1 \text{ s}^{-1}$ , as changing the value of  $\omega$  has no effect on the Fourier modes  $\alpha_n$ . Regarding the number of executions  $p$  of the circuit, we selected  $p = 375$ ,  $p = 1500$ , and  $p = 6000$  for the dimensions  $d = 16$ ,  $d = 32$ , and  $d = 64$ , respectively. The values of  $p$  were chosen to achieve roughly the same accuracy for the three values of  $d$ . Using PYTHON (version 3.11.3) with the SCIPY library (version 1.11.3), Fourier modes  $\alpha_n$  were calculated with Simpson’s rule for the numerical integration of Eq. (17). Due to the substantial size of the quantum circuit for the three dimensions analyzed in our simulations, we present the circuit for a lower dimension,  $d = 4$ , purely for illustrative purposes. This simplified example is shown in Fig. 2, allowing us to convey the structure without the complexity of the larger dimensions.

### VI. CONCLUSIONS

Concluding, this work presented a qubit-based quantum algorithm for prime number identification, rooted in the analysis of entangled subsystem dynamics. By employing a bipartite Hamiltonian and analyzing the Fourier modes of the reduced purity, we distinguish between prime and composite numbers within the range  $2 \leq n \leq 2(d - 1)$ . Implementing this on a qubit-based system involves transforming a two-qudit system into a qubit system, with the unitary gate of Eq. (2) implementable in polynomial time, contrary to the expected exponential gate requirements.

Our quantum circuit executes in three stages with a total gate cost indicating quadratic scaling in the number of digits of  $N = 2(d - 1)$ . Despite idealized simulations, implementation on quantum hardware is feasible but faces challenges such as qubit connectivity. Alternatives like trapped ion quantum computers or modified gate preparation and reduced purity measurement methods could overcome these.

The efficient realization of unitary operations demonstrates the potential for broader application in quantum computing, suggesting future work could extend this algorithm to verify larger primes. This progress in quantum algorithm optimization could significantly impact the field’s practical application to fundamental computational problems.

The data that support the findings of this study are available at [38]. This repository includes the PYTHON code for implementing the quantum algorithm in QISKIT, the simulation results, and auxiliary codes that support the theoretical findings.

### ACKNOWLEDGMENTS

This work was supported by the Coordination for the Improvement of Higher Education Personnel (CAPES), Grant No. 23081.031640/2023-17, by the National Council for Scientific and Technological Development (CNPq), Grants No. 309862/2021-3, No. 409673/2022-6, and No. 421792/2022-1, and the National Institute for the Science and Technology of Quantum Information (INCT-IQ), Grant No. 465469/2014-0. We thank A. D. Ribeiro for valuable discussions on the subject of this article.

### APPENDIX A: DIAGONAL UNITARY GATE IMPLEMENTATION USING WALSH FUNCTIONS

In this Appendix, we provide an overview of the algorithm introduced in Ref. [21] for implementing unitary operations on quantum computers. To begin, we establish some definitions. Let  $q$  denote the number of qubits and consider positive integers  $j$  and  $k$  with binary and dyadic representations given by

$$\text{bin}(j) = (j_q j_{q-1} \cdots j_1), \quad (\text{A1})$$

$$\text{dyad}(k) = (k_1 k_2 \cdots k_q), \quad (\text{A2})$$

where the most significant bit (MSB) is on the left. Henceforth, we assume  $j = 0, 1, \dots, 2^q - 1$  and  $k = 0, 1, \dots, 2^q - 1$ .

Next, we define the discrete Paley-ordered Walsh functions  $w_{jk}$  as

$$w_{jk} = (-1)^{\sum_{i=1}^q j_i k_i}. \quad (\text{A3})$$

Let us discretize the interval  $0 \leq x < 1$  into points given by

$$x_k = \frac{k}{2^q}. \quad (\text{A4})$$

Since the Walsh functions form an orthonormal basis, we can define the Walsh-Fourier transform for a function  $f_k = f(x_k)$  as follows:

$$a_j = \frac{1}{2^q} \sum_{k=0}^{2^q-1} f_k w_{jk}, \quad (\text{A5})$$

$$f_k = \sum_{j=0}^{2^q-1} a_j w_{jk}. \quad (\text{A6})$$

In qubit-based quantum computing, the state of  $q$  qubits generally takes the form  $|\psi\rangle = \sum_{k=0}^{2^q-1} c_k |k\rangle$ , where the computational basis  $|k\rangle$  is defined as

$$|k\rangle = |k_1 k_2 \cdots k_q\rangle, \quad (\text{A7})$$

with  $k$  represented in dyadic form  $\text{dyad}(k) = (k_1 k_2 \cdots k_q)$ . Now, let us define the unitary operator  $\hat{U} = e^{i\hat{f}}$  [29], where  $\hat{f}$  is a diagonal operator in the computational basis:

$$\hat{f}|k\rangle = f_k |k\rangle. \quad (\text{A8})$$

Walsh operators  $\{\hat{w}_j\}_{j=0}^{2^q-1}$  acting on  $q$  qubits are naturally defined as

$$\hat{w}_j = (\hat{Z}_1)^{j_1} \otimes (\hat{Z}_2)^{j_2} \otimes \cdots \otimes (\hat{Z}_q)^{j_q}, \quad (\text{A9})$$

where  $(\hat{Z}_i)^1 = \hat{Z}_i$  represents the Pauli Z operator and  $(\hat{Z}_i)^0 = \hat{I}$  denotes the identity matrix, both acting on the  $i$ th qubit  $q_i$ .

This definition of Walsh operators is advantageous because their action on the computational basis is given by

$$\hat{w}_j |k\rangle = w_{jk} |k\rangle. \quad (\text{A10})$$

This implies that the eigenvalues of Walsh operators  $\hat{w}_j$  are the Walsh functions  $w_{jk}$  and these operators form a basis for diagonal operators  $\hat{f}$ . Additionally, due to their form, Walsh operators commute. Therefore, considering  $\hat{w}_0 = \hat{I}$ , we can disregard  $j = 0$ , leading to the expression

$$\hat{U} = e^{i\hat{f}} = \prod_{j=1}^{2^q-1} e^{i a_j \hat{w}_j}. \quad (\text{A11})$$

In essence, to apply the method outlined in Ref. [21], we begin by determining the  $f_k$  values associated with the unitary gate  $\hat{U}$ . Subsequently, we construct the Walsh functions  $w_{jk}$  using the procedure described in Appendix C. With these components in hand, Eq. (A5) allows us to compute the Walsh angles  $a_j$ . Finally, utilizing the identity presented in Appendix B to construct the  $\hat{w}_j$  operators in Eq. (A11) yields the desired unitary  $\hat{U}$  with a gate cost of  $O(2^q)$  in general. This gate cost can be optimized by reordering the commuting exponential operators in Eq. (A11) using the GRAY code. It is important to note that, even with optimal construction, the quantum circuit for this method typically requires  $O(2^q)$  gates. However, as we will demonstrate in Appendix D, for the specific case of the  $q$ -qubit unitary gate  $\hat{U}(t)$  described in Eq. (2), implementation with a polynomial gate cost is achievable by identifying the null Walsh angles  $a_j(t)$ .

### APPENDIX B: RELATION BETWEEN PAULI Z GATES AND CNOT STAIRCASES

In this Appendix, we delve into a fundamental identity pivotal to our analysis, which concerns the tensor product of Pauli  $\hat{Z}$  operators. This identity plays a crucial role in simplifying the representation of quantum states and operations within our framework. To lay the groundwork for our discussion, we introduce the following essential notation and concepts.

$h_j$ , the Hamming weight of  $j$ , represents the number of 1's in the binary representation of  $j$ , corresponding to the number of  $\hat{Z}$  operators in the tensor product.

The identity operator  $\hat{I}^{\otimes r}$  acts on  $r$  qubits, serving as a placeholder in tensor products where no operation is performed.

The operators  $\hat{A}_{h_j}$  are constructed from a sequence of controlled-NOT ( $\widehat{\text{CNOT}}$ ) gates, defined as  $\hat{A}_{h_j} = \widehat{\text{CNOT}}_{h_j}^1 \widehat{\text{CNOT}}_{h_j}^2 \cdots \widehat{\text{CNOT}}_{h_j}^{h_j-1}$ , where  $\widehat{\text{CNOT}}_b^a$  denotes a  $\widehat{\text{CNOT}}$  gate with qubit  $q_a$  as the control and qubit  $q_b$  as the target.

With these definitions in place, we establish the following identity:

$$\hat{Z}_1 \otimes \hat{Z}_2 \otimes \cdots \otimes \hat{Z}_{h_j-1} \otimes \hat{Z}_{h_j} = \hat{A}_{h_j} (\hat{I}^{\otimes (h_j-1)} \otimes \hat{Z}) \hat{A}_{h_j}^{-1}. \quad (\text{B1})$$

This identity demonstrates how a tensor product of  $\hat{Z}$  operators can be equivalently expressed through a transformation involving  $\hat{A}_{h_j}$  and its inverse, significantly simplifying the representation and manipulation of such operations. Building upon this foundation, we further examine its implications in

the exponential form:

$$e^{ia_j(\hat{Z}_1 \otimes \hat{Z}_2 \otimes \dots \otimes \hat{Z}_{h_j-1} \otimes \hat{Z}_{h_j})} = \hat{A}_{h_j}(\hat{I}^{\otimes(h_j-1)} \otimes e^{ia_j \hat{Z}}) \hat{A}_{h_j}^{-1}. \quad (\text{B2})$$

This expression further underscores the utility of the  $\hat{A}_{h_j}$  transformation in facilitating the implementation of the exponential quantum gates  $e^{ia_j \hat{w}_j}$ .

Now, we proceed with the proofs. For the calculations below, unless otherwise convenient, we do not specify the qubit index  $i$  of Pauli operators  $\hat{Z}_i$  or any other operators. We start by rewriting the left side of Eq. (B1) using the projectors  $\hat{\Pi}_0 = |0\rangle\langle 0|$  and  $\hat{\Pi}_1 = |1\rangle\langle 1|$ :

$$\begin{aligned} & \hat{Z} \otimes \hat{Z} \otimes \dots \otimes \hat{Z} \otimes \hat{Z} \\ &= (\hat{\Pi}_0 - \hat{\Pi}_1) \otimes (\hat{\Pi}_0 - \hat{\Pi}_1) \otimes \dots \otimes (\hat{\Pi}_0 - \hat{\Pi}_1) \otimes \hat{Z} \\ &= \sum_{\text{bin}(s)} (-1)^{h_s} \hat{\Pi}_{s_1} \otimes \hat{\Pi}_{s_2} \otimes \dots \otimes \hat{\Pi}_{s_{(h_j-1)}} \otimes \hat{Z} \\ &= \sum_{\substack{\text{bin}(s) \\ h_s \text{ even}}} \hat{\Pi}_{s_1} \otimes \hat{\Pi}_{s_2} \otimes \dots \otimes \hat{\Pi}_{s_{(h_j-1)}} \otimes \hat{Z} \\ &\quad - \sum_{\substack{\text{bin}(s) \\ h_s \text{ odd}}} \hat{\Pi}_{s_1} \otimes \hat{\Pi}_{s_2} \otimes \dots \otimes \hat{\Pi}_{s_{(h_j-1)}} \otimes \hat{Z}, \end{aligned} \quad (\text{B3})$$

where the sum on  $\text{bin}(s)$  concerns all the possible binary representations  $\text{bin}(s) = (s_{(h_j-1)} s_{(h_j-2)} \dots s_1)$  of  $h_j - 1$  bits. It will be helpful to define

$$\Sigma_{\text{even}} = \sum_{\substack{\text{bin}(s) \\ h_s \text{ even}}} \hat{\Pi}_{s_1} \otimes \hat{\Pi}_{s_2} \otimes \dots \otimes \hat{\Pi}_{s_{(h_j-1)}}, \quad (\text{B4})$$

$$\Sigma_{\text{odd}} = \sum_{\substack{\text{bin}(s) \\ h_s \text{ odd}}} \hat{\Pi}_{s_1} \otimes \hat{\Pi}_{s_2} \otimes \dots \otimes \hat{\Pi}_{s_{(h_j-1)}}. \quad (\text{B5})$$

For these two definitions, the following relations are inherited from the projectors:

$$\Sigma_{\text{even}} \Sigma_{\text{even}} = \Sigma_{\text{even}}, \quad (\text{B6})$$

$$\Sigma_{\text{odd}} \Sigma_{\text{odd}} = \Sigma_{\text{odd}}, \quad (\text{B7})$$

$$\Sigma_{\text{even}} \Sigma_{\text{odd}} = \Sigma_{\text{odd}} \Sigma_{\text{even}} = 0. \quad (\text{B8})$$

Therefore, after defining  $\hat{I}$  as the identity gate acting on a single qubit and recalling that  $\hat{X} \hat{Z} \hat{X} = -\hat{Z}$ , we obtain

$$\begin{aligned} & \hat{Z} \otimes \hat{Z} \otimes \dots \otimes \hat{Z} \otimes \hat{Z} \\ &= \Sigma_{\text{even}} \otimes \hat{Z} - \Sigma_{\text{odd}} \otimes \hat{Z} \\ &= \Sigma_{\text{even}} \otimes \hat{Z} + \Sigma_{\text{odd}} \otimes \hat{X} \hat{Z} \hat{X} \\ &= (\Sigma_{\text{even}} \otimes \hat{I})(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z})(\Sigma_{\text{even}} \otimes \hat{I}) \\ &\quad + (\Sigma_{\text{odd}} \otimes \hat{X})(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z})(\Sigma_{\text{odd}} \otimes \hat{X}) \\ &= (\Sigma_{\text{even}} \otimes \hat{I})(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z})(\Sigma_{\text{even}} \otimes \hat{I}) \\ &\quad + (\Sigma_{\text{odd}} \otimes \hat{X})(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z})(\Sigma_{\text{odd}} \otimes \hat{X}) \\ &\quad + (\Sigma_{\text{odd}} \otimes \hat{X})(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z})(\Sigma_{\text{even}} \otimes \hat{I}) \\ &\quad + (\Sigma_{\text{even}} \otimes \hat{I})(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z})(\Sigma_{\text{odd}} \otimes \hat{X}) \\ &= (\Sigma_{\text{even}} \otimes \hat{I} + \Sigma_{\text{odd}} \otimes \hat{X})(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z}) \\ &\quad \times (\Sigma_{\text{even}} \otimes \hat{I} + \Sigma_{\text{odd}} \otimes \hat{X}). \end{aligned} \quad (\text{B9})$$

To continue, we examine the product of two controlled-NOT gates targeting the same qubit:

$$\begin{aligned} \widehat{\text{CNOT}}_3^1 \widehat{\text{CNOT}}_3^2 &= (\hat{\Pi}_0 \otimes \hat{I} \otimes \hat{I} + \hat{\Pi}_1 \otimes \hat{I} \otimes \hat{X}) \\ &\quad \times (\hat{I} \otimes \hat{\Pi}_0 \otimes \hat{I} + \hat{I} \otimes \hat{\Pi}_1 \otimes \hat{X}) \\ &= \hat{\Pi}_0 \otimes \hat{\Pi}_0 \otimes \hat{I} + \hat{\Pi}_0 \otimes \hat{\Pi}_1 \otimes \hat{X} \\ &\quad + \hat{\Pi}_1 \otimes \hat{\Pi}_0 \otimes \hat{X} + \hat{\Pi}_1 \otimes \hat{\Pi}_1 \otimes \hat{I} \\ &= (\hat{\Pi}_0 \otimes \hat{\Pi}_0 + \hat{\Pi}_1 \otimes \hat{\Pi}_1) \otimes \hat{I} \\ &\quad + (\hat{\Pi}_1 \otimes \hat{\Pi}_0 + \hat{\Pi}_0 \otimes \hat{\Pi}_1) \otimes \hat{X}. \end{aligned} \quad (\text{B10})$$

The equation above suggests a similar form for a more general case. In fact, it holds that

$$\begin{aligned} \hat{A}_{h_j} &= \widehat{\text{CNOT}}_{h_j}^1 \widehat{\text{CNOT}}_{h_j}^2 \dots \widehat{\text{CNOT}}_{h_j}^{h_j-1} \\ &= \sum_{\substack{\text{bin}(s) \\ h_s \text{ even}}} \hat{\Pi}_{s_1} \otimes \hat{\Pi}_{s_2} \otimes \dots \otimes \hat{\Pi}_{s_{(h_j-1)}} \otimes \hat{I} \\ &\quad + \sum_{\substack{\text{bin}(s) \\ h_s \text{ odd}}} \hat{\Pi}_{s_1} \otimes \hat{\Pi}_{s_2} \otimes \dots \otimes \hat{\Pi}_{s_{(h_j-1)}} \otimes \hat{X} \\ &= \Sigma_{\text{even}} \otimes \hat{I} + \Sigma_{\text{odd}} \otimes \hat{X}. \end{aligned} \quad (\text{B11})$$

Then, because  $\hat{A}_{h_j}^{-1} = \hat{A}_{h_j}$ , we obtain the proposed expression (B1) by using the identity (B11) on Eq. (B9):

$$\hat{Z}_1 \otimes \hat{Z}_2 \otimes \dots \otimes \hat{Z}_{h_j-1} \otimes \hat{Z}_{h_j} = \hat{A}_{h_j}(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z}) \hat{A}_{h_j}^{-1}. \quad (\text{B12})$$

Using this result, we can further demonstrate the validity of Eq. (B2):

$$\begin{aligned} e^{ia_j(\hat{Z}_1 \otimes \hat{Z}_2 \otimes \dots \otimes \hat{Z}_{h_j-1} \otimes \hat{Z}_{h_j})} &= \sum_{n=0}^{\infty} \frac{(ia_j)^n}{n!} (\hat{Z} \otimes \hat{Z} \otimes \dots \otimes \hat{Z})^n \\ &= \sum_{n=0}^{\infty} \frac{(ia_j)^n}{n!} [\hat{A}_{h_j}(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z}) \hat{A}_{h_j}^{-1}]^n \\ &= \sum_{n=0}^{\infty} \frac{(ia_j)^n}{n!} \hat{A}_{h_j}(\hat{I}^{\otimes(h_j-1)} \otimes \hat{Z}^n) \hat{A}_{h_j}^{-1} \\ &= \hat{A}_{h_j} \left( \hat{I}^{\otimes(h_j-1)} \otimes \sum_{n=0}^{\infty} \frac{(ia_j \hat{Z})^n}{n!} \right) \hat{A}_{h_j}^{-1} \\ &= \hat{A}_{h_j}(\hat{I}^{\otimes(h_j-1)} \otimes e^{ia_j \hat{Z}}) \hat{A}_{h_j}^{-1}. \end{aligned} \quad (\text{B13})$$

In this context, we revisit the formulation of Walsh operators, as delineated in Eq. (A9), represented by  $\hat{w}_j = (\hat{Z}_1)^{j_1} \otimes (\hat{Z}_2)^{j_2} \otimes \dots \otimes (\hat{Z}_q)^{j_q}$ , where the action of  $e^{ia_j \hat{w}_j}$  on a  $q$ -qubit basis state  $|k\rangle = |k_1 k_2 \dots k_q\rangle$  is considered. To elaborate on the analysis, we introduce a strategic reordering of the indices  $j_i$ , segregating them into two distinct sets: the first, denoted by  $\{m_i\}_{i=1}^{h_j}$ , corresponds to indices where  $j_{m_i} = 1$ , spanning the initial  $h_j$  bits; the latter set,  $\{m_i\}_{i=h_j+1}^q$ , encompasses indices with  $j_{m_i} = 0$ , accounting for the remaining  $q - h_j$  bits. The accordingly reconfigured states of qubits  $q_i$  and the operators  $(\hat{Z}_i)^{j_i}$  can be achieved by applying SWAP gates. This culmi-

nates in the revised Walsh operator  $\hat{\Omega}_j$  and revised basis state  $|k'\rangle$ , articulated as

$$\hat{\Omega}_j = (\hat{Z}_{m_1} \otimes \hat{Z}_{m_2} \otimes \cdots \otimes \hat{Z}_{m_{h_j}}) \otimes (\hat{I}_{m_{h_j+1}} \otimes \hat{I}_{m_{h_j+2}} \otimes \cdots \otimes \hat{I}_{m_q}), \quad (\text{B14})$$

$$|k'\rangle = |k_{m_1} k_{m_2} \cdots k_{m_q}\rangle. \quad (\text{B15})$$

Leveraging Eq. (B13), we have

$$\begin{aligned} e^{ia_j \hat{\Omega}_j} |k'\rangle &= e^{ia_j (\hat{Z}_{m_1} \otimes \hat{Z}_{m_2} \otimes \cdots \otimes \hat{Z}_{m_{h_j}})} \otimes (\hat{I}_{m_{h_j+1}} \otimes \hat{I}_{m_{h_j+2}} \otimes \cdots \otimes \hat{I}_{m_q}) \\ &\quad \times |k_{m_1} k_{m_2} \cdots k_{m_q}\rangle \\ &= e^{ia_j (\hat{Z}_{m_1} \otimes \hat{Z}_{m_2} \otimes \cdots \otimes \hat{Z}_{m_{h_j}})} \\ &\quad \otimes (\hat{I}_{m_{h_j+1}} \otimes \hat{I}_{m_{h_j+2}} \otimes \cdots \otimes \hat{I}_{m_q}) |k_{m_1} k_{m_2} \cdots k_{m_q}\rangle \\ &= [\hat{A}_{h_j} (\hat{I}^{\otimes (h_j-1)} \otimes e^{ia_j \hat{Z}}) \hat{A}_{h_j}^{-1}] \\ &\quad \otimes \hat{I}^{\otimes (q-h_j)} |k_{m_1} k_{m_2} \cdots k_{m_q}\rangle. \end{aligned} \quad (\text{B16})$$

As it was stated previously, our objective lies in the action of the original operator  $e^{ia_j \hat{w}_j}$  on the original basis state  $|k\rangle$ , i.e.,  $e^{ia_j \hat{w}_j} |k\rangle$ . However, through the application of the same SWAP gates used before to Eq. (B16), we restore the original sequence of qubits and Pauli operators, thereby preserving the structural integrity of the action of the operator  $e^{ia_j \hat{w}_j}$  on the basis state  $|k\rangle$ .

### APPENDIX C: WALSH MATRICES CONSTRUCTION

A well-established result in the literature states that any discrete Walsh function  $w_{jk}$  can be represented as a product of Rademacher functions, with the exception of  $w_{0k}$ , which trivially remains a constant function  $w_{0k} = 1$ . Rademacher functions, denoted as  $w_{l_r k}$ , are Walsh functions where  $l_r$  is a power of 2, specifically  $l_r = 2^{(r-1)}$ .

To illustrate, consider a Walsh matrix  $w_{(\dots)}$ , where  $w_{jk} = w_{(j,k)}$ , representing Paley-ordered Walsh functions arranged in a square matrix of dimensions  $2^q \times 2^q$ . In matrix formalism, any row  $w_{(j,\dots)}$  can be understood as the columnwise product of the corresponding  $h_j$  Rademacher rows  $\{w_{(l_{m_i}, \dots)}\}_{i=1}^{h_j}$ , organized by ascending order of magnitude  $l_{m_1} < l_{m_2} < \cdots < l_{m_{h_j}}$ . Here,  $h_j$  signifies the Hamming weight of  $j$ . The Rademacher rows  $w_{(l_{m_i}, \dots)}$  satisfy  $j = \sum_{i=1}^{h_j} l_{m_i} = \sum_{i=1}^{h_j} 2^{(m_i-1)}$ .

Here, we introduce this result as a proposition with minor adjustments. This adaptation enables us to introduce the required notation for the theorem detailed in the next Appendix, which concerns the implementation of  $\hat{U}(t)$ . Consistent with convention, we index the rows and columns of our Walsh matrix starting from 0, with the maximum index value being  $2^q - 1$ .

*Proposition 1.* Consider a positive integer  $r$  satisfying  $1 \leq r \leq q$ , where  $l_r = 2^{(r-1)}$ . Let  $\{m_i\}_{i=1}^{h_j}$  be a sequence of integers in increasing order of magnitude and let  $j$  be any row index of the  $2^q \times 2^q$  Walsh matrix  $w_{(\dots)}$ . Then, the row  $w_{(j,\dots)}$  will satisfy only one of the following statements

(1) If  $j = l_r$ , then the Rademacher row  $w_{(j,\dots)} = w_{(l_r,\dots)}$  of the Walsh matrix is given by

$$w_{(l_r,\dots)} = [(R_r)(-R_r)(R_r)(-R_r) \cdots (R_r)(-R_r)], \quad (\text{C1})$$

where  $R_r = (+1)(\times T_r)$  is a row of length  $T_r = 2^{(q-r)}$  and  $T_r$  is called the period of the row  $w_{(l_r,\dots)}$ . The notation means that  $w_{(l_r,\dots)}$  is composed of sign alternating sequences of  $T_r$  elements. These sequences are  $R_r$  and  $-R_r$  and all their elements are, respectively, equal to  $+1$  and  $-1$ .

(2) If  $j \neq l_r$ , we define  $j = \sum_{i=1}^{h_j} l_{m_i} = \sum_{i=1}^{h_j} 2^{(m_i-1)}$ ; then the row  $w_{(j,\dots)}$  of the Walsh matrix is given by

$$w_{(j,\dots)} = [(R_{m_1})(-R_{m_1})(R_{m_1})(-R_{m_1}) \cdots (R_{m_1})(-R_{m_1})], \quad (\text{C2})$$

where  $R_{m_i}$  is obtained from the recursive relation

$$R_{m_{(q-1)}} = [(R_{m_i})(-R_{m_i})(R_{m_i})(-R_{m_i}) \cdots (R_{m_i})(-R_{m_i})], \quad (\text{C3})$$

with  $R_{m_0} = w_{(j,\dots)}$  and  $R_{m_{h_j}} = (+1)(\times T_{m_{h_j}})$ . Each  $R_{m_i}$ , for  $1 \leq i \leq h_j$ , is a row of length  $T_{m_i} = 2^{(q-m_i)}$ .

*Proof.* We shall prove each case of Proposition 1 separately.

1. *Case*  $j = l_r$ . The binary representation of  $j = l_r = 2^{(r-1)}$  is given by

$$\text{bin}(j) = (000 \cdots 01_r 0 \cdots 000), \quad (\text{C4})$$

where we have introduced the notation  $1_r = j_r = 1$  to make it clear that the only nonzero binary element of  $\text{bin}(j)$  is in the position  $r$ .

*Definition 1.* We define the exponent  $S(j, k)$  of Eq. (A3) as

$$S(j, k) = \sum_{i=1}^q j_i k_i. \quad (\text{C5})$$

If  $S(j, k)$  is even, then  $w_{(j,k)} = 1$ . If  $S(j, k)$  is odd, then  $w_{(j,k)} = -1$ .

Then, for any  $0 \leq k \leq 2^q - 1$ , we have

$$k_r = 0 \Rightarrow S(j, k) = j_r k_r = \text{even}, \quad (\text{C6})$$

$$k_r = 1 \Rightarrow S(j, k) = j_r k_r = \text{odd}. \quad (\text{C7})$$

Now, we consider only a partial dyadic representation string  $\text{dyad}_r(k)$  of  $k$ . Since the dyadic string of  $k$  is the same as its reverse binary string, a partial dyadic string up to the position  $r$  is defined as

$$\text{dyad}_r(k) = (k_r k_{r+1} \cdots k_{(q-1)} k_q). \quad (\text{C8})$$

There are  $q - r + 1$  entries in the partial dyadic string of Eq. (C8), resulting in a total of  $2^{(q-r+1)}$  combinations. The first half of these combinations corresponds to  $k_r = 0$  and the second half corresponds to  $k_r = 1$ . That is, the first  $2^{(q-r)}$  elements of  $S(j, \cdot)$  are even and the next  $2^{(q-r)}$  elements of  $S(j, \cdot)$  are odd. Note that we did not consider the other  $r - 1$  entries that show up in the complete dyadic string  $\text{dyad}(k)$  of  $k$ . This is justified by the fact that the pattern we just described will remain true for any configuration of the disregarded  $r - 1$  entries, as changing any of these entries allows another total of  $2^{(q-r+1)}$  possible combinations with the same pattern for  $\text{dyad}_r(k)$ . From this, we infer that  $S(j, \cdot)$  is completely defined



by alternating sequences of even and odd elements, with each sequence having length  $T_r = 2^{(q-r)}$ . Thus, using that  $w_{(j,k)} = (-1)^{S(j,k)}$ , the Rademacher row  $w_{(j,\cdot)} = w_{(l_r,\cdot)}$  will be given by

$$w_{(l_r,\cdot)} = [(R_r)(-R_r)(R_r)(-R_r) \cdots (R_r)(-R_r)], \quad (\text{C9})$$

with  $R_r = (+1)(\times T_r)$  and period  $T_r = 2^{(q-r)}$ .

2. *Case  $j \neq l_r$ .* Even though  $j$  here cannot be written as a power of 2, it can always be written as a sum of these powers:

$$j = \sum_{i=1}^{h_j} l_{m_i} = \sum_{i=1}^{h_j} 2^{(m_i-1)}, \quad (\text{C10})$$

and we conveniently choose the indices  $m_i$  to match the positions of the nonzero binary elements of  $j$ , i.e.,  $j_{m_i} = 1$ . That is, if we consider only the nonzero binary elements  $\text{bin}_{\neq 0}(j)$  of the binary representation  $\text{bin}(j)$  of  $j$ , then

$$\text{bin}_{\neq 0}(j) = (j_{m_{h_j}} \cdots j_{m_2} j_{m_1}). \quad (\text{C11})$$

Thus, for any  $0 \leq k \leq 2^q - 1$ , we have

$$S(j, k) = \sum_{i=1}^q j_i k_i = \sum_{i=1}^{h_j} j_{m_i} k_{m_i}. \quad (\text{C12})$$

Because each term  $j_{m_i} k_{m_i}$  in the sum above is the corresponding  $l_{m_i}$  term  $S(l_{m_i}, k)$ , it must be true that

$$\begin{aligned} w_{(j,k)} &= (-1)^{\sum_{i=1}^{h_j} S(l_{m_i}, k)} \\ &= \prod_{i=1}^{h_j} (-1)^{S(l_{m_i}, k)} \\ &= \prod_{i=1}^{h_j} w_{(l_{m_i}, k)}. \end{aligned} \quad (\text{C13})$$

Equation (C13) means that any row  $w_{(j,\cdot)}$  is the columnwise product of the corresponding Rademacher rows  $w_{(l_{m_i}, \cdot)}$  such that  $j = \sum_{i=1}^{h_j} l_{m_i}$ .

Since  $l_{m_1} < l_{m_2} < \cdots < l_{m_{h_j}}$ , the respective periods  $T_{m_i} = 2^{(q-m_i)}$ , for  $1 \leq i \leq h_j$ , must obey  $T_{m_1} > T_{m_2} > \cdots > T_{m_{h_j}}$ . We know that, for  $0 \leq k \leq T_{m_1} - 1$ , we have  $w_{(l_{m_1}, k)} = +1$ . Then, if  $0 \leq k \leq T_{m_1} - 1$ , the elements  $w_{(j,k)}$  of Eq. (C13) can be written as  $w_{(j,k)} = \prod_{i=2}^{h_j} w_{(l_{m_i}, k)}$ . Now, any Rademacher row  $w_{(l_{m_i}, \cdot)}$  has a period  $T_{m_i}$  and thus is composed of alternating pairs of sequences, where each pair is made of a sequence of  $T_{m_i}$  terms all equal to +1 and a sequence of  $T_{m_i}$  terms all equal to -1. This implies that the number of times  $L_{[1,i]}$  we can fit these pairs of  $2T_{m_i}$  terms into the length of the largest period  $T_{m_1}$  is given by

$$\begin{aligned} L_{[1,i]} &= \frac{T_{m_1}}{2T_{m_i}} \\ &= \frac{2^{(q-m_1)}}{2 \times 2^{(q-m_i)}} \\ &= \frac{2^{(m_i-m_1)}}{2} \\ &= \frac{l_{m_i}}{2l_{m_1}}, \end{aligned} \quad (\text{C14})$$

which evidently shows, for  $i \geq 2$ , that  $L_{[1,i]}$  is an integer, specifically, a power of 2. Now, for  $T_{m_1} \leq k \leq 2T_{m_1} - 1$  we must have  $w_{(l_{m_1}, k)} = -1$  and the pattern of  $w_{(j,\cdot)}$  will occur again for these next  $T_{m_1}$  values of  $k$ , except that in this case we get a sign change, i.e.,  $w_{(j,k)} = -\prod_{i=2}^{h_j} w_{(l_{m_i}, k)}$ . After these first  $2T_{m_1}$  terms, the periodicity of  $w_{(l_{m_1}, \cdot)}$  implies that the aforementioned pattern will continue to repeat itself until we have the row  $w_{(j,\cdot)}$  completely filled. From that, we conclude that  $T_{m_1}$  is a common period for every composing Rademacher row of  $w_{(j,\cdot)}$ . We can also analyze how many pairs of  $2T_{m_i}$  terms fit into the length of any other period  $T_{m_{i'}} > T_{m_i}$ , that is, calculate the expression for  $L_{[i',i]}$ . In this case, for any  $i > i'$ , we have

$$\begin{aligned} L_{[i',i]} &= \frac{T_{m_{i'}}}{2T_{m_i}} \\ &= \frac{2^{(q-m_{i'})}}{2 \times 2^{(q-m_i)}} \\ &= \frac{2^{(m_i-m_{i'})}}{2} \\ &= \frac{l_{m_i}}{2l_{m_{i'}}}. \end{aligned} \quad (\text{C15})$$

This shows that any period  $T_{m_{i'}}$ , for  $h_j - 1 \geq i' \geq 1$ , has a length that can be perfectly fit by an integer number of pairs of  $2T_{m_i}$  terms such that  $T_{m_{i'}} > T_{m_i}$ . The extreme case of  $T_{m_{h_j}}$ , however, will never have any lower period, a relevant fact that leads to the following definition.

*Definition 2.* To proceed, for  $1 \leq i \leq h_j$ , we define the recursive relation

$$R_{m_{(i-1)}} = [(R_{m_i})(-R_{m_i})(R_{m_i})(-R_{m_i}) \cdots (R_{m_i})(-R_{m_i})], \quad (\text{C16})$$

where  $R_{m_{h_j}} = (+1)(\times T_{m_{h_j}})$ , and  $R_{m_0}$  and  $R_{m_i}$  have respective periods  $T_{m_0} = 2^q$  and  $T_{m_i} = 2^{(q-m_i)}$ .

Utilizing the recursive relation in Eq. (C16), we demonstrate that  $w_{(j,\cdot)} = R_{m_0}$ . This recursive relation maintains a critical connection with the Rademacher rows  $w_{(l_{m_i}, \cdot)}$ . The initial condition  $R_{m_{h_j}} = (+1)(\times T_{m_{h_j}})$  is chosen due to  $T_{m_{h_j}}$  being the minimal period, thus serving as the recursive sequence's base case. For periods  $T_{m_{(i-1)}} > T_{m_{h_j}}$ , the sequence iteratively incorporates  $R_{m_i}$ , further integrating the effects of  $R_{m_{(i+1)}}$  until the base case  $R_{m_{h_j}} = (+1)(\times T_{m_{h_j}})$  is reached. Particularly,  $R_{m_0}$  is synthesized through several  $(R_{m_1})(-R_{m_1})$  pairs, effectively encapsulating the periodic characteristics of the composing Rademacher rows of  $w_{(j,\cdot)}$ . Therefore, each term in the sequence  $R_{m_1}$  of  $T_{m_1}$  terms corresponds to either  $\prod_{i=2}^{h_j} w_{(l_{m_i}, k)}$  or its negative counterpart,  $-\prod_{i=2}^{h_j} w_{(l_{m_i}, k)}$ , for the relevant  $T_{m_1}$  columns  $k$ . As we have shown, the products  $\pm \prod_{i=2}^{h_j} w_{(l_{m_i}, k)}$  fully compose the row  $w_{(j,\cdot)}$ , leading to the conclusion that  $w_{(j,\cdot)} = R_{m_0}$ . Hence we have

$$w_{(j,\cdot)} = [(R_{m_1})(-R_{m_1})(R_{m_1})(-R_{m_1}) \cdots (R_{m_1})(-R_{m_1})]. \quad (\text{C17})$$

### APPENDIX D: DEMONSTRATION OF POLYNOMIAL COST

As we have shown in the first Appendix, to use the method of Ref. [21], we must calculate the Walsh angles  $a_j$ , given by

$$\begin{aligned} a_j &= \frac{1}{2^q} \sum_{k=0}^{2^q-1} f_k w_{jk} \\ &= \frac{1}{2^q} [w_{(j,\cdot)}] \times [\vec{f}]^T, \end{aligned} \quad (\text{D1})$$

where we have introduced the symbol  $T$  for transposition, since  $\vec{f}$  is defined as a row vector. The components  $f_k$  of  $\vec{f}$  are extracted as the eigenvalues of the operator

$$\hat{f} = \sum_{k=0}^{2^q-1} f_k |k\rangle \langle k|. \quad (\text{D2})$$

Then, a general unitary operator of the form  $\hat{U} = e^{i\hat{f}}$ , when represented in the computational basis  $|k\rangle$ , is given by the

$$\vec{f} = [(1, 2, 3, \dots, d), (2, 4, 6, \dots, 2d), (3, 6, 9, \dots, 3d), (4, 8, 12, \dots, 4d), \dots, (d, 2d, 3d, \dots, d^2)]. \quad (\text{D5})$$

There is also another global factor  $\frac{1}{2^q} = \frac{1}{d^2}$  in the expression (D1), which can be equally disregarded in the calculations. Taking into account these two global factors,  $a_j$  will be redefined to be

$$a_j = [w_{(j,\cdot)}] \times [\vec{f}]^T, \quad (\text{D6})$$

with  $\vec{f}$  given by Eq. (D5). Thus the relevant Walsh angles  $a_j(t)$  for  $\hat{U}(t)$  should be defined as

$$a_j(t) = \frac{-\omega t}{d^2} a_j. \quad (\text{D7})$$

Since there is no risk of confusion,  $a_j$  and  $a_j(t)$  are both referred to as ‘‘Walsh angles’’ in this work. The following theorem concerns the derivations of the expressions for the nonzero Walsh angles  $a_j(t)$  corresponding to the unitary gate  $\hat{U}(t)$  of Eq. (2). Even though we are using  $q$  even here, it should be noted that a similar theorem holds for odd values of  $q$ , opening possibilities for the efficient implementation of any  $2^q \times 2^q$  diagonal unitary gate  $\hat{U}(t)$  with the form of Eq. (2).

*Theorem 2.* Let  $W$  be the set of nonzero Walsh angles  $a_j(t)$  for the  $q$ -qubit unitary gate  $\hat{U}(t)$  of Eq. (2), with  $q$  even. Then

$$W = W_1 \cup W_2, \quad (\text{D8})$$

with

$$W_1 = \left\{ a_j(t) = \begin{cases} \frac{d(1+d)\omega t}{8j} & \text{if } j \leq d/2, \\ \frac{d^2(1+d)\omega t}{8j} & \text{if } j \geq d, \end{cases} \left| h_j = 1 \right. \right\}, \quad (\text{D9})$$

following diagonal matrix:

$$U = \begin{bmatrix} D_1 & 0 & \dots & 0 \\ 0 & D_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & D_d \end{bmatrix}, \quad (\text{D3})$$

where  $d^2 = 2^q$ , with  $d$  a power of 2, and the  $D_s$  are diagonal matrices for  $1 \leq s \leq d$ . In this paper, we define these diagonal matrices  $D_s$  as

$$D_s = \begin{bmatrix} e^{i\varphi_s} & 0 & \dots & 0 \\ 0 & e^{2i\varphi_s} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{di\varphi_s} \end{bmatrix}, \quad (\text{D4})$$

where  $i^2 = -1$ ,  $\varphi_s = s\kappa$ , and  $\kappa$  is a real number. Comparing with the unitary gate  $\hat{U}(t)$  of Eq. (2) for our quantum algorithm, we must use  $\kappa = -\omega t$ . However, since  $\kappa$  is a global factor for  $\vec{f}$  and thus a multiplicative factor in the expression for  $a_j$  in Eq. (D1), we can choose  $\kappa = 1$  to facilitate further calculations. The vector  $\vec{f}$  that we are going to use is then given by

and

$$W_2 = \left\{ a_j(t) = \frac{-d^3 \omega t}{16l_{m_1} l_{m_2}} \left| h_j = 2 \right. \right. \\ \left. \left. \text{and } j = l_{m_1} + l_{m_2} \text{ for } l_{m_1} \leq d/2 \text{ and } l_{m_2} \geq d \right. \right\}, \quad (\text{D10})$$

where  $h_j$  is the Hamming weight of  $j$  and  $d^2 = 2^q$ .

The quantum circuit corresponding to the implementation of  $\hat{U}(t)$  is presented for  $d = 4$  in Fig. 3. As it is evident, Theorem 2 eliminates the vast majority of the  $2^q - 1$  Walsh angles  $a_j(t)$  that would be necessary, in general, to implement this type of unitary gate exactly. In Lemma 3, we show that the number of gates needed to exactly implement  $\hat{U}(t)$  is a polynomial function of  $q$ .

In what follows, we list some definitions and their respective properties, in order to use them in the proof of Theorem 2, that will be presented afterwards.

*Definition 3.* Elementary vectors are defined as

$$\vec{P}_\eta = [\eta, 2\eta, 3\eta, \dots, d\eta]. \quad (\text{D11})$$

*Definition 4.* Extended vectors are defined as

$$\vec{P}_{\{\alpha|\alpha+\beta\}} = [\vec{P}_\alpha, \vec{P}_{\alpha+1}, \vec{P}_{\alpha+2}, \dots, \vec{P}_{\alpha+\beta}]. \quad (\text{D12})$$

*Definition 5.* Partial vectors are defined as

$$\vec{P}_{\{\alpha|\alpha+\beta\}}^{(\eta)} = [P_\alpha^{(\eta)}, P_{\alpha+1}^{(\eta)}, P_{\alpha+2}^{(\eta)}, \dots, P_{\alpha+\beta}^{(\eta)}], \quad (\text{D13})$$

where we defined the notation  $\vec{P}_\eta = [P_1^{(\eta)}, P_2^{(\eta)}, P_3^{(\eta)}, \dots, P_d^{(\eta)}]$  for elementary vectors.

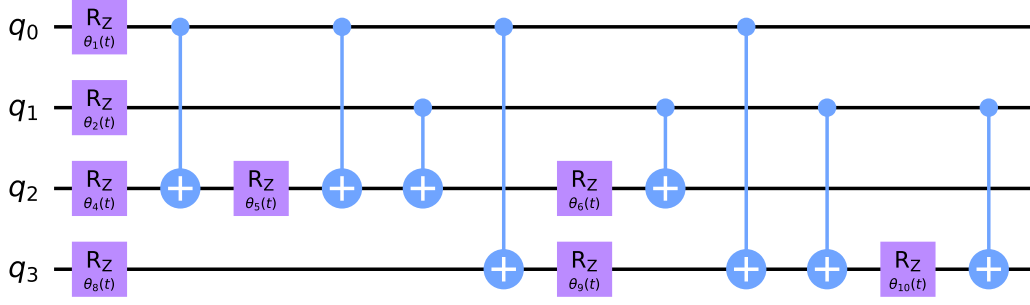


FIG. 3. Quantum circuit that implements the unitary gate  $\hat{U}(t)$  for  $d = 4$ . We used the mathematical identity demonstrated in Appendix B to express an exponential of Z Pauli gates as a single Z rotation  $\hat{R}_z(\theta_j(t))$ , with the angles given by  $\theta_j(t) = -2a_j(t)$ , and several controlled-NOT gates.

*Property 6.* Using Eq. (D5) for  $\vec{f}$  and Definition 3 for  $\vec{P}_\eta$ , we can write

$$\vec{f} = [\vec{P}_1, \vec{P}_2, \vec{P}_3, \dots, \vec{P}_d]. \quad (\text{D14})$$

*Property 7.* By the Definition 3 for elementary vectors  $\vec{P}_\eta$ , it follows that

$$\vec{P}_\eta = \eta \vec{P}_1. \quad (\text{D15})$$

*Property 8.* It follows directly from Property 7 that

$$\vec{P}_\eta - \vec{P}_{\eta+\Delta} = -\Delta \vec{P}_1. \quad (\text{D16})$$

*Property 9.* Applying Property 8 for each elementary vector of Definition 4 gives us

$$\begin{aligned} & \vec{P}_{|\alpha|\alpha+\beta} - \vec{P}_{|\alpha+\Delta|\alpha+\Delta+\beta} \\ &= [\vec{P}_\alpha, \vec{P}_{\alpha+1}, \vec{P}_{\alpha+2}, \dots, \vec{P}_{\alpha+\beta}] \\ & \quad - [\vec{P}_{\alpha+\Delta}, \vec{P}_{\alpha+\Delta+1}, \vec{P}_{\alpha+\Delta+2}, \dots, \vec{P}_{\alpha+\Delta+\beta}] \\ &= [\vec{P}_\alpha - \vec{P}_{\alpha+\Delta}, \vec{P}_{\alpha+1} - \vec{P}_{\alpha+\Delta+1}, \vec{P}_{\alpha+2} \\ & \quad - \vec{P}_{\alpha+\Delta+2}, \dots, \vec{P}_{\alpha+\beta} - \vec{P}_{\alpha+\Delta+\beta}] \\ &= -\Delta[\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1]. \end{aligned} \quad (\text{D17})$$

*Property 10.* It follows, respectively, from Definition 5 of partial vectors  $\vec{p}_{|\alpha|\alpha+\beta}^{(\eta)}$ , Property 7, and Definition 3 applied to  $\vec{P}_1$  that

$$\begin{aligned} & \vec{p}_{|\alpha|\alpha+\beta}^{(\eta)} - \vec{p}_{|\alpha+\Delta|\alpha+\Delta+\beta}^{(\eta)} \\ &= [p_\alpha^{(\eta)}, p_{\alpha+1}^{(\eta)}, p_{\alpha+2}^{(\eta)}, \dots, p_{\alpha+\beta}^{(\eta)}] \\ & \quad - [p_{\alpha+\Delta}^{(\eta)}, p_{\alpha+\Delta+1}^{(\eta)}, p_{\alpha+\Delta+2}^{(\eta)}, \dots, p_{\alpha+\Delta+\beta}^{(\eta)}] \\ &= [p_\alpha^{(\eta)} - p_{\alpha+\Delta}^{(\eta)}, p_{\alpha+1}^{(\eta)} - p_{\alpha+\Delta+1}^{(\eta)}, p_{\alpha+2}^{(\eta)} - p_{\alpha+\Delta+2}^{(\eta)}, \\ & \quad \dots, p_{\alpha+\beta}^{(\eta)} - p_{\alpha+\Delta+\beta}^{(\eta)}] \\ &= \eta [p_\alpha^{(1)} - p_{\alpha+\Delta}^{(1)}, p_{\alpha+1}^{(1)} - p_{\alpha+\Delta+1}^{(1)}, p_{\alpha+2}^{(1)} - p_{\alpha+\Delta+2}^{(1)}, \\ & \quad \dots, p_{\alpha+\beta}^{(1)} - p_{\alpha+\Delta+\beta}^{(1)}] \\ &= -\eta \Delta [1, 1, 1, \dots, 1]. \end{aligned} \quad (\text{D18})$$

Building upon the definitions and properties outlined earlier, we now introduce an important vector, denoted by  $\vec{f}_{m_i}^\sigma$ , constructed from components of  $\vec{f}$ . This vector emerges as a cornerstone of our analysis, serving as the basis for deriving

key properties that will be extensively used in our proof. Alongside  $\vec{f}_{m_i}^\sigma$ , we define a scalar quantity,  $F_i^\sigma$ , designed to facilitate subsequent calculations.

*Definition 11.* Let  $i$  and  $\sigma$  be two integers such that  $1 \leq i \leq h_j$  and  $1 \leq \sigma \leq l_{m_i}$ , where  $l_{m_i} = \frac{d^2}{2T_{m_i}}$ . We define  $\vec{f}_{m_i}^\sigma$  as the vector formed by the  $\sigma$ th  $2T_{m_i}$  elements of  $\vec{f}$ .

*Definition 12.* We use the recursive relation (C16) and Definition 11 to define the scalar  $F_i^\sigma = [(R_{m_i})(-R_{m_i})] \times [\vec{f}_{m_i}^\sigma]^T$ .

*Property 13.* If we use Definitions 11 and 12 with  $i = 1$ , then

$$\begin{aligned} a_j &= [w_{(j,\cdot)}] \times [\vec{f}]^T \\ &= [R_{m_0}] \times [\vec{f}]^T \\ &= [(R_{m_1})(-R_{m_1})(R_{m_1})(-R_{m_1}) \dots (R_{m_1})(-R_{m_1})] \\ & \quad \times [\vec{f}_{m_1}^1, \vec{f}_{m_1}^2, \vec{f}_{m_1}^3, \dots, \vec{f}_{m_1}^\sigma, \dots, \vec{f}_{m_1}^{l_{m_1}}]^T \\ &= \sum_{\tau=1}^{l_{m_1}} [(R_{m_1})(-R_{m_1})] \times [\vec{f}_{m_1}^\tau]^T \\ &= \sum_{\tau=1}^{l_{m_1}} F_1^\tau. \end{aligned} \quad (\text{D19})$$

The following two properties form the foundation of our strategy to distinguish which Walsh angles  $a_j$  are null and which are not. The technique we will use involves determining whether the composing powers  $l_{m_i}$  of  $j$  possess corresponding periods  $T_{m_i} \geq d$  or  $T_{m_i} \leq d/2$ . Depending on this categorization, we will then employ either extended or partial vector notation.

*Property 14.* Let  $T_{m_i} = \mu_i d$ , for  $\mu_i \geq 1$  a power of 2. Then,  $\vec{f}_{m_i}^\tau = [\vec{P}_{\{(2\tau-2)\mu_i+1|(2\tau-1)\mu_i\}}, \vec{P}_{\{(2\tau-1)\mu_i+1|2\tau\mu_i\}}]$  and after using Definition 12 and Property 9

$$\begin{aligned} F_i^\tau &= [(R_{m_i})(-R_{m_i})][\vec{P}_{\{(2\tau-2)\mu_i+1|(2\tau-1)\mu_i\}}, \vec{P}_{\{(2\tau-1)\mu_i+1|2\tau\mu_i\}}]^T \\ &= [R_{m_i}] \times [\vec{P}_{\{(2\tau-2)\mu_i+1|(2\tau-1)\mu_i\}}]^T \\ & \quad + [-R_{m_i}] \times [\vec{P}_{\{(2\tau-1)\mu_i+1|2\tau\mu_i\}}]^T \\ &= [R_{m_i}][\vec{P}_{\{(2\tau-2)\mu_i+1|(2\tau-1)\mu_i\}} - \vec{P}_{\{(2\tau-1)\mu_i+1|2\tau\mu_i\}}]^T \\ &= [R_{m_i}] \times [-\mu_i(\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T. \end{aligned} \quad (\text{D20})$$

Thus, by defining  $F_1^\tau = F_1$ , we have the following property for expression (D19):

$$a_j = l_{m_1} F_1. \quad (\text{D21})$$

*Property 15.* Let  $T_{m_i} = d/v_i$ , for  $v_i \geq 2$  a power of 2. Then,  $\vec{f}_{m_i}^\tau = [\vec{P}_{\{(2\tau-2)d/v_i+1\}(2\tau-1)d/v_i}^{(\eta)}, \vec{P}_{\{(2\tau-1)d/v_i+1\}2\tau d/v_i}^{(\eta)}]$ . After using Definition 12 and Property 10, we obtain

$$\begin{aligned} F_i^\tau &= [(R_{m_i})(-R_{m_i})] \\ &\quad \times [\vec{P}_{\{(2\tau-2)d/v_i+1\}(2\tau-1)d/v_i}^{(\eta)}, \vec{P}_{\{(2\tau-1)d/v_i+1\}2\tau d/v_i}^{(\eta)}]^T \\ &= [R_{m_i}] \times [\vec{P}_{\{(2\tau-2)d/v_i+1\}(2\tau-1)d/v_i}^{(\eta)}]^T \\ &\quad + [-R_{m_i}] \times [\vec{P}_{\{(2\tau-1)d/v_i+1\}2\tau d/v_i}^{(\eta)}]^T \\ &= [R_{m_i}] [\vec{P}_{\{(2\tau-2)d/v_i+1\}(2\tau-1)d/v_i}^{(\eta)} - \vec{P}_{\{(2\tau-1)d/v_i+1\}2\tau d/v_i}^{(\eta)}]^T \\ &= [R_{m_i}] \times \left[ -\frac{\eta d}{v_i} (1, 1, 1, \dots, 1) \right]^T. \end{aligned} \quad (\text{D22})$$

We know that  $1 \leq \eta \leq d$  and, within a fixed  $\eta$ , there is  $\frac{d}{T_{m_1}}$  partial vectors of the form  $\vec{P}_{\{\alpha|\alpha+T_{m_1}-1\}}^{(\eta)}$ . Then, we define  $F_1^\tau = F_1^{(\eta)}$  and replace the sum of Eq. (D19) on  $\tau$  with the sum on  $\eta$  and a multiplication by the term  $\frac{d}{2T_{m_1}} = \frac{v_1}{2}$  to obtain the following property:

$$a_j = \frac{v_1}{2} \sum_{\eta=1}^d F_1^{(\eta)}. \quad (\text{D23})$$

*Proof.* Having stated these definitions and properties, now we go to the proof of the theorem. We shall prove separately the formulas of  $a_j(t)$  for each Hamming weight  $h_j = 1$ ,  $h_j = 2$ , and  $h_j \geq 3$ .

1. *Case  $h_j = 1$ .* In this case, we have  $j = l_{m_1} = 2^{m_1-1}$  and  $R_{m_1} = (+1)(\times T_{m_1})$ .

1.1. *Subcase  $l_{m_1} \leq d/2$ .* Here, we have  $T_{m_1} = \mu_1 d$ , for  $\mu_1 \geq 1$  a power of 2. First, we calculate the scalar  $F_1$  using Eq. (D20):

$$\begin{aligned} F_1 &= [R_{m_1}] \times [-\mu_1(\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T \\ &= [(+1)(\times T_{m_1})] \times [-\mu_1(\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T \\ &= -\mu_1(P_1 + P_1 + P_1 + \dots + P_1) \\ &= -\mu_1^2 P_1, \end{aligned} \quad (\text{D24})$$

where we have defined  $P_1$  as the sum of all the elements of  $\vec{P}_1$ , that is,  $P_1 = 1 + 2 + 3 + \dots + d$ . Because  $P_1 = \frac{(1+d)d}{2}$ , then

$$F_1 = -\frac{\mu_1^2(1+d)d}{2}. \quad (\text{D25})$$

Now, since  $T_{m_1} = \mu_1 d$  and by definition  $T_{m_1} = d^2/2l_{m_1}$ , with  $l_{m_1} = j$ , we can write  $\mu_1 = d/2j$ . If we use Property 14 to calculate  $a_j$ , then

$$\begin{aligned} a_j &= l_{m_1} F_1 \\ &= j \left( -\frac{\mu_1^2(1+d)d}{2} \right) \\ &= \frac{-(1+d)d^3}{8j}. \end{aligned} \quad (\text{D26})$$

To obtain the relevant Walsh angle  $a_j(t)$ , we make use of Eq. (D7). Thus

$$a_j(t) = \frac{d(1+d)\omega t}{8j}. \quad (\text{D27})$$

1.2. *Subcase  $l_{m_1} \geq d$ .* For this subcase, we have  $T_{m_1} = d/v_1$  for  $v_1 \geq 2$  a power of 2. The scalar  $F_1^{(\eta)} = F_1^\tau$  is calculated by the expression (D22):

$$\begin{aligned} F_1^{(\eta)} &= [R_{m_1}] \times \left[ -\frac{\eta d}{v_1} (1, 1, 1, \dots, 1) \right]^T \\ &= [(+1)(\times T_{m_1})] \times \left[ -\frac{\eta d}{v_1} (1, 1, 1, \dots, 1) \right]^T \\ &= \frac{-\eta d}{v_1} (1 + 1 + 1 + \dots + 1) \\ &= \frac{-\eta d^2}{v_1^2}. \end{aligned} \quad (\text{D28})$$

Using that  $T_{m_1} = d/v_1 = d^2/2l_{m_1}$ , with  $l_{m_1} = j$ , we write  $v_1 = 2j/d$ . Then, using Property 15 to calculate the Walsh angles  $a_j$ ,

$$\begin{aligned} a_j &= \frac{v_1}{2} \sum_{\eta=1}^d F_1^{(\eta)} \\ &= \frac{-v_1 d^2}{2v_1^2} \sum_{\eta=1}^d \eta \\ &= \frac{-(1+d)d^4}{8j}. \end{aligned} \quad (\text{D29})$$

Therefore, by Eq. (D7),

$$a_j(t) = \frac{d^2(1+d)\omega t}{8j}. \quad (\text{D30})$$

Thus, for  $h_j = 1$ , we always have

$$a_j(t) \neq 0. \quad (\text{D31})$$

2. *Case  $h_j = 2$ .* In this case, we have  $j = l_{m_1} + l_{m_2}$ . We use the recursive relation of Eq. (C16) to write

$$R_{m_1} = [(R_{m_2})(-R_{m_2})(R_{m_2})(-R_{m_2}) \dots (R_{m_2})(-R_{m_2})], \quad (\text{D32})$$

with  $R_{m_2} = (+1)(\times T_{m_2})$ .

2.1. *Subcase  $l_{m_1} < d/2$ ,  $l_{m_2} \leq d/2$ .* Here, we have  $T_{m_1} = \mu_1 d$  and  $T_{m_2} = \mu_2 d$ , where both  $\mu_1 \geq 2$  and  $\mu_2 \geq 1$  are powers of 2. First, we calculate the scalar  $F_1$  using Eq. (D20):

$$\begin{aligned} F_1 &= [R_{m_1}] \times [-\mu_1(\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T \\ &= [(R_{m_2})(-R_{m_2})(R_{m_2})(-R_{m_2}) \dots (R_{m_2})(-R_{m_2})] \\ &\quad \times [-\mu_1(\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T \\ &= \frac{-\mu_1 T_{m_1}}{2T_{m_2}} ([R_{m_2}] \times (\vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)^T - [R_{m_2}] \\ &\quad \times (\vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)^T) \\ &= 0, \end{aligned} \quad (\text{D33})$$

where the factor  $T_{m_1}/2T_{m_2}$  was introduced because we collapsed the sum of all the  $T_{m_1}/2T_{m_2}$  products  $[(R_{m_2})(-R_{m_2})] \times$

$[\vec{P}_1, \vec{P}_1, \dots, \vec{P}_1]^T$  into just a single product. Then, using Property 14 for  $a_j$ , we obtain

$$\begin{aligned} a_j &= l_{m_1} F_1 \\ &= 0, \end{aligned} \quad (\text{D34})$$

and by Eq. (D7)

$$a_j(t) = 0. \quad (\text{D35})$$

2.2. *Subcase  $l_{m_1} \geq d, l_{m_2} > d$ .* In this subcase, we have  $T_{m_1} = d/v_1$  and  $T_{m_2} = d/v_2$ , where both  $v_1 \geq 2$  and  $v_2 \geq 4$  are powers of 2. Now, we calculate the scalar  $F_1^{(\eta)}$  using Eq. (D22),

$$\begin{aligned} F_1^{(\eta)} &= [R_{m_1}] \times \left[ \frac{-\eta d}{v_1} (1, 1, 1, \dots, 1) \right]^T \\ &= [(R_{m_2})(-R_{m_2})(R_{m_2})(-R_{m_2}) \cdots (R_{m_2})(-R_{m_2})] \\ &\quad \times \left[ \frac{-\eta d}{v_1} (1, 1, 1, \dots, 1) \right]^T \\ &= \frac{-\eta d T_{m_1}}{2v_1 T_{m_2}} ([R_{m_2}] \times (1, 1, \dots, 1)^T - [R_{m_2}] \\ &\quad \times (1, 1, \dots, 1)^T) \\ &= 0, \end{aligned} \quad (\text{D36})$$

for any  $1 \leq \eta \leq d$ . From Property 15 for  $a_j$ , we conclude that

$$\begin{aligned} a_j &= \frac{v_1}{2} \sum_{\eta=1}^d F_1^{(\eta)} \\ &= 0. \end{aligned} \quad (\text{D37})$$

Thus, by Eq. (D7), we must also have

$$a_j(t) = 0. \quad (\text{D38})$$

2.3. *Subcase  $l_{m_1} \leq d/2, l_{m_2} \geq d$ .* For this final subcase, we have  $T_{m_1} = \mu_1 d$  and  $T_{m_2} = d/v_2$ , where  $\mu_1 \geq 1$  and  $v_2 \geq 2$  are powers of 2. We start by calculating the scalar  $F_1$  using Eq. (D20):

$$\begin{aligned} F_1 &= [R_{m_1}] \times [-\mu_1 (\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T \\ &= [(R_{m_2})(-R_{m_2})(R_{m_2})(-R_{m_2}) \cdots (R_{m_2})(-R_{m_2})] \\ &\quad \times [-\mu_1 (\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T. \end{aligned} \quad (\text{D39})$$

However, we can no longer calculate products of the form  $[R_{m_2}] \times (\vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)^T$ , because now  $R_{m_2}$  has length  $T_{m_2} = d/v_2$ , which is less than the length of  $\vec{P}_1$ . To be able to calculate the right side of expression (D39), we break each  $\vec{P}_1$  into smaller parts of length  $T_{m_2}$ , using partial vectors  $\vec{p}_{\{\alpha + T_{m_2} - 1\}}^{(\eta)}$ .

First, we notice that  $\vec{P}_1$  corresponds to partial vectors with  $\eta = 1$ . Second, since  $\vec{P}_1$  has length  $d$ , we can fit  $d/T_{m_2} = v_2$  partial vectors  $\vec{p}_{\{\alpha + T_{m_2} - 1\}}^{(\eta)}$  into  $\vec{P}_1$ . That is, considering Eq. (D22), the  $v_2/2$  products that we have to calculate are related to  $F_2^\tau = F_2^{(1)}$  by the expression

$$\begin{aligned} -\mu_1 F_2^{(1)} &= [(R_{m_2})(-R_{m_2})] [-\mu_1 (\vec{p}_{\{(2\tau-2)d/v_2+1\}}^{(1)} \\ &\quad \vec{p}_{\{(2\tau-1)d/v_2+1\}}^{(1)})]^T \\ &= [R_{m_2}] \times \left[ \frac{\mu_1 d}{v_2} (1, 1, \dots, 1) \right]^T. \end{aligned} \quad (\text{D40})$$

After calculating the product of Eq. (D40) and multiplying it by the number  $v_2/2$ , we should then multiply the result by the number of elementary vectors  $\vec{P}_1$  appearing in Eq. (D39), which is  $T_{m_1}/d = \mu_1$ . Therefore,

$$\begin{aligned} F_1 &= (\mu_1) \left( \frac{v_2}{2} \right) (-\mu_1 F_2^{(1)}) \\ &= [R_{m_2}] \times \left[ \frac{\mu_1^2 d}{2} (1, 1, \dots, 1) \right]^T \\ &= [(+1)(\times T_{m_2})] \times \left[ \frac{\mu_1^2 d}{2} (1, 1, \dots, 1) \right]^T \\ &= \frac{\mu_1^2 d}{2} (1 + 1 + \dots + 1) \\ &= \frac{\mu_1^2 d^2}{2v_2}. \end{aligned} \quad (\text{D41})$$

With  $F_1$  calculated,  $a_j$  will be given by Property 14. To simplify the final result for  $a_j$ , we will use  $T_{m_1} = \mu_1 d = d^2/2l_{m_1}$  and  $T_{m_2} = d/v_2 = d^2/2l_{m_2}$  to write  $\mu_1 = d/2l_{m_1}$  and  $v_2 = 2l_{m_2}/d$ . Therefore, by Property 14 we have that

$$\begin{aligned} a_j &= l_{m_1} F_1 \\ &= l_{m_1} \left( \frac{\mu_1^2 d^2}{2v_2} \right) \\ &= \frac{d^5}{16l_{m_1} l_{m_2}}. \end{aligned} \quad (\text{D42})$$

Again, by making use of Eq. (D7), we obtain

$$a_j(t) = \frac{-d^3 \omega t}{16l_{m_1} l_{m_2}}. \quad (\text{D43})$$

Thus, for  $h_j = 2$ , if  $l_{m_1} \leq d/2$  and  $l_{m_2} \geq d$ , we necessarily have

$$a_j(t) \neq 0. \quad (\text{D44})$$

Otherwise  $a_j(t) = 0$ .

3. *Case  $h_j \geq 3$ .* For this case, we have in general  $j = l_{m_1} + l_{m_2} + \sum_{i=3}^{h_j} l_{m_i}$ .

3.1. *Subcase  $l_{m_1} < d/2, l_{m_2} \leq d/2$ .* Here, we have a similar situation to subcase 2.1:  $T_{m_1} = \mu_1 d$  and  $T_{m_2} = \mu_2 d$ . However, we also have other powers  $l_{m_i}$  with respective periods  $T_{m_i} = d^2/2l_{m_i}$  for  $i \geq 3$ . We will show that, for any such  $T_{m_i}$ , it must be true that  $a_j = 0$ . We recall the recursive relation (C16) and use Eq. (D20) to obtain

$$\begin{aligned} F_1 &= [R_{m_1}] \times [-\mu_1 (\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T \\ &= [(R_{m_2})(-R_{m_2})(R_{m_2})(-R_{m_2}) \cdots (R_{m_2})(-R_{m_2})] \\ &\quad \times [-\mu_1 (\vec{P}_1, \vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)]^T \\ &= \frac{-\mu_1 T_{m_1}}{2T_{m_2}} ([R_{m_2}] \times (\vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)^T - [R_{m_2}] \\ &\quad \times (\vec{P}_1, \vec{P}_1, \dots, \vec{P}_1)^T) \\ &= 0. \end{aligned} \quad (\text{D45})$$

In the scenario of subcase 2.1, we have  $R_{m_2} = (+1)(\times T_{m_2})$  and  $F_1 = 0$ . Although for the present subcase we have  $R_{m_2} \neq (+1)(\times T_{m_2})$ , the scalar  $F_1$  is again identically null, just like

in subcase 2.1. Note that we have not imposed any restrictions on  $l_{m_i}$ , that is,  $F_1 = 0$  regardless of whether  $l_{m_i} \leq d/2$  or  $l_{m_i} \geq d$ . Now, from Property 14 we have  $a_j = l_{m_1} F_1$ . Thus, if there are any others  $l_{m_i}$  such that  $j = l_{m_1} + l_{m_2} + \sum_{i=3}^{h_j} l_{m_i}$ , it holds that

$$a_j = 0. \quad (\text{D46})$$

Therefore, by Eq. (D7), it follows that

$$a_j(t) = 0. \quad (\text{D47})$$

**3.2. Sub-case  $l_{m_1} \geq d, l_{m_2} > d, l_{m_i} > d$ .** This situation is an extension of subcase 2.2:  $T_{m_1} = d/v_1$  and  $T_{m_2} = d/v_2$ . Again, we have powers  $l_{m_i}$  with respective periods  $T_{m_i}$ , with  $i \geq 3$ . By the recursive relation (C16) and Eq. (D22), we obtain

$$\begin{aligned} F_1^{(\eta)} &= [R_{m_1}] \times \left[ \frac{-\eta d}{v_1} (1, 1, 1, \dots, 1) \right]^T \\ &= [(R_{m_2})(-R_{m_2})(R_{m_2})(-R_{m_2}) \cdots (R_{m_2})(-R_{m_2})] \\ &\quad \times \left[ \frac{-\eta d}{v_1} (1, 1, 1, \dots, 1) \right]^T \\ &= \frac{-\eta d T_{m_1}}{2v_1 T_{m_2}} ([R_{m_2}] \times (1, 1, \dots, 1)^T - [R_{m_2}] \\ &\quad \times (1, 1, \dots, 1)^T) \\ &= 0, \end{aligned} \quad (\text{D48})$$

for any  $1 \leq \eta \leq d$ . Similarly to what happened for  $F_1$  in subcase 3.1, here the result of  $F_1^{(\eta)}$  is independent of the form of  $R_{m_2}$ . This is relevant because, in the present subcase,  $R_{m_2}$  is obtained by other  $R_{m_i}$  that we did not specify. Now, from Property 15 we have  $a_j = \frac{v_1}{2} \sum_{\eta=1}^d F_1^{(\eta)}$ . Then, it must be true that

$$a_j = 0 \quad (\text{D49})$$

and by Eq. (D7) it follows that

$$a_j(t) = 0. \quad (\text{D50})$$

**3.3. Subcase  $l_{m_1} \leq d/2, l_{m_2} \geq d, l_{m_i} > d$ .** In subcase 2.3, we have shown that  $a_j \neq 0$ . Here, since there are other powers  $l_{m_i}$ , the situation, however, will turn out to be different. We should repeat the calculations of subcase 2.3 by first using Eq. (D40):

$$-\mu_1 F_2^{(1)} = [R_{m_2}] \times \left[ \frac{\mu_1 d}{v_2} (1, 1, \dots, 1) \right]^T. \quad (\text{D51})$$

We must calculate  $F_1$ , since  $T_{m_1} \geq d$ . From the calculations of subcase 2.3,  $F_1$  is given by  $F_1 = (\mu_1)(\frac{v_2}{2})(-\mu_1 F_2^{(1)})$ . Thus

$$\begin{aligned} F_1 &= (\mu_1) \left( \frac{v_2}{2} \right) (-\mu_1 F_2^{(1)}) \\ &= [R_{m_2}] \times \left[ \frac{\mu_1^2 d}{2} (1, 1, \dots, 1) \right]^T \\ &= [(R_{m_3})(-R_{m_3})(R_{m_3})(-R_{m_3}) \cdots (R_{m_3})(-R_{m_3})] \\ &\quad \times \left[ \frac{\mu_1^2 d}{2} (1, 1, \dots, 1) \right]^T \end{aligned}$$

$$\begin{aligned} &= \frac{\mu_1^2 d T_{m_2}}{4T_{m_3}} ([R_{m_3}] \times (1, \dots, 1)^T - [R_{m_3}] \times (1, \dots, 1)^T) \\ &= 0. \end{aligned} \quad (\text{D52})$$

From Property 14, the Walsh angles for this subcase are given by  $a_j = l_{m_1} F_1$ . Therefore,

$$a_j = 0 \quad (\text{D53})$$

and from Eq. (D7) we get

$$a_j(t) = 0. \quad (\text{D54})$$

Just like it happened for subcase 2.2, where we had  $l_{m_1} \geq d$  and  $l_{m_2} > d$ , the Walsh angles  $a_j(t)$  are also null here. This is true for any  $l_{m_i \geq 3}$  that composes  $j$ . The conclusion is that, for any  $j$  with  $h_j \geq 3$ , we have  $a_j(t) = 0$ .

We know that the only nonzero Walsh angles  $a_j(t)$  are those with Hamming weight  $h_j = 1$  or  $h_j = 2$ , that is, cases 1 and 2. In case 1, it is always true that  $a_j(t) \neq 0$ . In case 2, the Walsh angles are  $a_j(t) \neq 0$  if and only if we simultaneously have  $l_{m_1} \leq d/2$  and  $l_{m_2} \geq d$ . Therefore, if  $W$  is the set of nonzero Walsh angles, then it is the union of two subsets  $W_1$  and  $W_2$ , composed, respectively, by the nonzero Walsh angles with  $h_j = 1$  and  $h_j = 2$ . We can summarize this as

$$W = W_1 \cup W_2, \quad (\text{D55})$$

with

$$W_1 = \left\{ a_j(t) = \begin{cases} \frac{d(1+d)\omega t}{8j} & \text{if } j \leq d/2, \\ \frac{d^2(1+d)\omega t}{8j} & \text{if } j \geq d, \end{cases} \middle| h_j = 1 \right\} \quad (\text{D56})$$

and

$$W_2 = \left\{ a_j(t) = \frac{-d^3 \omega t}{16l_{m_1} l_{m_2}} \middle| h_j = 2 \text{ and } j = l_{m_1} + l_{m_2}, \right. \\ \left. \text{where } l_{m_1} \leq d/2 \text{ and } l_{m_2} \geq d \right\}. \quad (\text{D57})$$

**Lemma 3.** Let  $G_2(q)$  be the number of gates necessary to implement the  $q$ -qubit unitary gate  $\hat{U}(t)$  exactly. If we use only  $Z$  rotations and controlled-NOT gates, then  $G_2(q) = \frac{3}{4}q^2 + q$ .

*Proof.* As it is established in Appendix B, we can calculate the exponential operators  $e^{ia_j(t)\hat{w}_j}$  in Eq. (A11) by applying a  $Z$  rotation on qubit  $q_{m_{h_j}}$ , where  $m_{h_j}$  is the MSB of  $j$  and two controlled-NOT gates targeted on  $q_{m_{h_j}}$  for each controlling qubit. That is, the number of gates for a single  $e^{ia_j(t)\hat{w}_j}$  is given by one  $Z$  rotation and  $2(h_j - 1)$  controlled-NOT gates, resulting in  $2h_j - 1$  gates. Now, from Theorem 2, the only nonzero Walsh angles  $a_j(t)$  are those for which we have  $h_j = 1$  or  $h_j = 2$ .

For  $h_j = 1$  it is always true that  $a_j(t) \neq 0$  and the respective values of  $j$  correspond to powers of 2. We know that, within  $1 \leq j \leq 2^q - 1$ , there are  $q$  powers of 2. Then, the total number of gates  $G_2^{(1)}(q)$  necessary for  $h_j = 1$  is  $G_2^{(1)}(q) = (2h_j - 1)q$ . Thus

$$G_2^{(1)}(q) = q. \quad (\text{D58})$$

For  $h_j = 2$ , we have  $a_j(t) \neq 0$  if and only if  $j = l_{m_1} + l_{m_2}$  with  $l_{m_1} \leq d/2$  and  $l_{m_2} \geq d$ . To find how many gates  $G_2^{(2)}(q)$

are needed here, we must count how many combinations of  $l_{m_1}$  and  $l_{m_2}$  are possible. First, we notice that  $d/2 = 2^{\frac{q}{2}}/2 = 2^{(\frac{q}{2}-1)}$ . If we pick  $l_{m_1}$  as satisfying  $l_{m_1} = d/2$ , then  $d/2 = l_{m_1} = 2^{(m_1-1)}$  implies that  $m_1 = \frac{q}{2}$ . That is, there are  $\frac{q}{2}$  possible values for  $l_{m_1} \leq d/2$ . Because there is a total of  $q$  powers of 2 in the interval  $1 \leq j \leq 2^q - 1$ , then there is also  $q - \frac{q}{2} = \frac{q}{2}$  possible values for  $l_{m_2}$ . We conclude that the number of combinations of  $l_{m_1}$  and  $l_{m_2}$  is  $(\frac{q}{2})(\frac{q}{2}) = \frac{q^2}{4}$ . Then, the number  $G_2^{(2)}(q)$  of gates necessary for  $h_j = 2$  is  $G_2^{(2)} = (2h_j - 1)\frac{q^2}{4}$ . Thus

$$G_2^{(2)} = \frac{3}{4}q^2. \quad (\text{D59})$$

Finally, the total gate cost  $G_2(q) = G_2^{(1)}(q) + G_2^{(2)}(q)$  for the implementation of the unitary  $\hat{U}(t)$  is

$$G_2(q) = \frac{3}{4}q^2 + q. \quad (\text{D60})$$

#### APPENDIX E: PURITY ESTIMATION USING A VARIATION OF THE SWAP TEST

In Ref. [26], the authors explore an interferometric setup to extract  $\text{Tr}(\hat{U}\hat{\rho})$  based on its correlation with the visibility  $v$ , where  $\hat{U}$  represents a unitary gate and  $\hat{\rho}$  denotes the density operator of the system. Their investigation draws parallels between this quantum circuit and the one employed in the SWAP test. However, there are notable distinctions: they utilize a controlled- $\hat{U}$  gate instead of a controlled-SWAP gate and consider density operators  $\hat{\rho}$  instead of pure states  $|\psi\rangle$ .

The authors further contend that, by selecting  $\hat{U}$  as the SWAP gate and letting  $\hat{\rho} = \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2}$  represent the joint density operator of two subsystems  $S_1$  and  $S_2$ , a specific scenario arises:

$$\text{Tr}(\hat{\rho}_{S_1}\hat{\rho}_{S_2}) = 2P_0 - 1, \quad (\text{E1})$$

where  $P_0$  signifies the probability of measuring the state  $|0\rangle$  for the ancilla qubit subsequent to the application of the unitary gates forming the quantum circuit.

In this Appendix, we present an operational proof for Eq. (E1). Particularly, when  $\hat{\rho}_{S_1} = \hat{\rho}_{S_2}$ , it yields the purity  $\text{Tr}(\hat{\rho}_{S_1}^2) = 2P_0 - 1$ , a key quantity for our quantum algorithm. Before delving into the proof, we first introduce an identity pertaining to the SWAP gate.

*Proposition 4.* Let  $\hat{V}_1$  and  $\hat{V}_2$  be two linear operators acting, respectively, on  $d^2$ -dimensional Hilbert spaces  $H_1$  and  $H_2$ , with  $H_1 = H_2$ . Then, the following identity for the  $\widehat{\text{SWAP}}$  gate holds

$$\text{Tr}((\hat{V}_1 \otimes \hat{V}_2)\widehat{\text{SWAP}}) = \text{Tr}(\hat{V}_1\hat{V}_2). \quad (\text{E2})$$

*Proof.* We will calculate both sides of Eq. (E2) and show that they lead to the same expression. To do that, we start by defining the matrix representations of  $\hat{V}_i$  on the computational basis:

$$\hat{V}_i = \begin{bmatrix} V_i^{(1,1)} & V_i^{(1,2)} & \dots & V_i^{(1,d)} \\ V_i^{(2,1)} & V_i^{(2,2)} & \dots & V_i^{(2,d)} \\ \vdots & \vdots & \ddots & \vdots \\ V_i^{(d,1)} & V_i^{(d,2)} & \dots & V_i^{(d,d)} \end{bmatrix}. \quad (\text{E3})$$

Then, we can write

$$\hat{V}_1\hat{V}_2 = \begin{bmatrix} \sum_{k=1}^d V_1^{(1,k)}V_2^{(k,1)} & \dots & * \\ \vdots & \ddots & \vdots \\ * & * & \sum_{k=1}^d V_1^{(d,k)}V_2^{(k,d)} \end{bmatrix}, \quad (\text{E4})$$

with  $*$  denoting matrix elements we do not need. Thus

$$\text{Tr}(\hat{V}_1\hat{V}_2) = \sum_{j,k=1}^d V_1^{(j,k)}V_2^{(k,j)}. \quad (\text{E5})$$

Now, we calculate the left side of Eq. (E2):

$$\begin{aligned} & \text{Tr}((\hat{V}_1 \otimes \hat{V}_2)\widehat{\text{SWAP}}) \\ &= \text{Tr}\left(\left(\sum_{j,k=1}^d V_1^{(j,k)}|j\rangle\langle k| \otimes \sum_{l,m=1}^d V_2^{(l,m)}|l\rangle\langle m|\right)\widehat{\text{SWAP}}\right) \\ &= \sum_{j,k=1}^d \sum_{l,m=1}^d V_1^{(j,k)}V_2^{(l,m)}\text{Tr}(|j\rangle \otimes |l\rangle\langle k| \otimes \langle m|\widehat{\text{SWAP}}) \\ &= \sum_{j,k=1}^d \sum_{l,m=1}^d V_1^{(j,k)}V_2^{(l,m)}\text{Tr}(|j\rangle \otimes |l\rangle\langle m| \otimes \langle k|) \\ &= \sum_{j,k=1}^d \sum_{l,m=1}^d V_1^{(j,k)}V_2^{(l,m)}\langle m|j\rangle\langle k|l\rangle \\ &= \sum_{j,k=1}^d V_1^{(j,k)}V_2^{(k,j)}. \end{aligned} \quad (\text{E6})$$

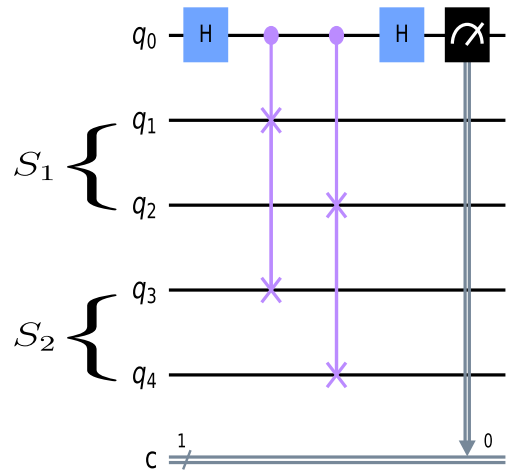


FIG. 4. Schematic representation of a modified SWAP test employed to determine  $\text{Tr}(\hat{\rho}_{S_1}\hat{\rho}_{S_2})$  in a four-qubit system ( $q = 4$ ). Initially, qubits  $q_1$  and  $q_2$  are prepared in the state  $\hat{\rho}_{S_1}$ , while qubits  $q_3$  and  $q_4$  are prepared in the state  $\hat{\rho}_{S_2}$ . An ancilla qubit  $q_0$  is introduced. The circuit involves a Hadamard gate applied to  $q_0$ , qubit-qubit controlled-SWAP gates between the two qubit sets, and a final Hadamard gate on  $q_0$ . By measuring  $q_0$  multiple times and obtaining  $P_0$ , the quantity  $\text{Tr}(\hat{\rho}_{S_1}\hat{\rho}_{S_2})$  can be estimated as  $2P_0 - 1$ .

Therefore,

$$\text{Tr}((\hat{V}_1 \otimes \hat{V}_2)\widehat{\text{SWAP}}) = \text{Tr}(\hat{V}_1 \hat{V}_2). \quad (\text{E7})$$

We are now prepared to demonstrate the validity of Eq. (E1). This proof will be conducted by constructing the proposed quantum circuit introduced in Ref. [26], illustrated in Fig. 4 for the specific scenario involving  $q = 4$  qubits and one ancilla qubit. Our system comprises an ancilla qubit  $q_0$  and two subsystems, denoted as  $S_1$  and  $S_2$ . Initially, the system's density operator  $\hat{\rho}$  is given by the tensor product:

$$\hat{\rho} = |0\rangle\langle 0| \otimes \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2}. \quad (\text{E8})$$

Following the sequence of gates outlined in the quantum circuit, we define  $\hat{\rho}^{(1)} = \hat{H}\hat{\rho}\hat{H}$ ,  $\hat{\rho}^{(2)} = (\widehat{\text{CSWAP}})(\hat{\rho}^{(1)})(\widehat{\text{CSWAP}})$ , and  $\hat{\rho}^{(3)} = \hat{H}\hat{\rho}^{(2)}\hat{H}$ . Commencing with  $\hat{\rho}^{(1)}$ , we proceed by applying a Hadamard gate  $\hat{H}$  to  $q_0$ :

$$\begin{aligned} \hat{\rho}^{(1)} &= \hat{H}\hat{\rho}\hat{H} \\ &= \frac{1}{2}(|0\rangle + |1\rangle)(\langle 0| + \langle 1|) \otimes \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2} \\ &= |+\rangle\langle +| \otimes \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2}, \end{aligned} \quad (\text{E9})$$

with  $|\pm\rangle := \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$ . By applying the controlled-SWAP gate  $\widehat{\text{CSWAP}}$ , we obtain

$$\begin{aligned} \hat{\rho}^{(2)} &= (\widehat{\text{CSWAP}})(\hat{\rho}^{(1)})(\widehat{\text{CSWAP}}) \\ &= (|0\rangle\langle 0| \otimes \hat{I} + |1\rangle\langle 1| \otimes \widehat{\text{SWAP}})(|+\rangle\langle +| \otimes \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2}) \\ &\quad \times (|0\rangle\langle 0| \otimes \hat{I} + |1\rangle\langle 1| \otimes \widehat{\text{SWAP}}) \\ &= \frac{1}{\sqrt{2}}(|0\rangle\langle +| \otimes \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2} + |1\rangle\langle +| \otimes \widehat{\text{SWAP}}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})) \\ &\quad \times (|0\rangle\langle 0| \otimes \hat{I} + |1\rangle\langle 1| \otimes \widehat{\text{SWAP}}) \\ &= \frac{1}{2}(|0\rangle\langle 0| \otimes \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2} + |1\rangle\langle 0| \otimes \widehat{\text{SWAP}}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})) \\ &\quad + |0\rangle\langle 1| \otimes (\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})\widehat{\text{SWAP}} \\ &\quad + |1\rangle\langle 1| \otimes \widehat{\text{SWAP}}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})\widehat{\text{SWAP}}. \end{aligned} \quad (\text{E10})$$

To finalize, we apply another Hadamard gate to  $q_0$ :

$$\begin{aligned} \hat{\rho}^{(3)} &= \hat{H}\hat{\rho}^{(2)}\hat{H} \\ &= \frac{1}{2}(|+\rangle\langle +| \otimes \hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2} + |-\rangle\langle +| \otimes \widehat{\text{SWAP}}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})) \\ &\quad + |+\rangle\langle -| \otimes (\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})\widehat{\text{SWAP}} \\ &\quad + |-\rangle\langle -| \otimes \widehat{\text{SWAP}}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})\widehat{\text{SWAP}}. \end{aligned} \quad (\text{E11})$$

The next step is to calculate the reduced density operator  $\hat{\rho}_0$  for the ancilla qubit. To do that, we take the partial trace over  $S_1$  and  $S_2$ :

$$\begin{aligned} \hat{\rho}_0 &= \text{Tr}_{S_1 S_2}(\hat{\rho}^{(3)}) \\ &= \frac{1}{2}(|+\rangle\langle +| \text{Tr}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2}) + |-\rangle\langle +| \text{Tr}[\widehat{\text{SWAP}}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})]) \\ &\quad + |+\rangle\langle -| \text{Tr}[(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})\widehat{\text{SWAP}}] \\ &\quad + |-\rangle\langle -| \text{Tr}[\widehat{\text{SWAP}}(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})\widehat{\text{SWAP}}] \end{aligned}$$

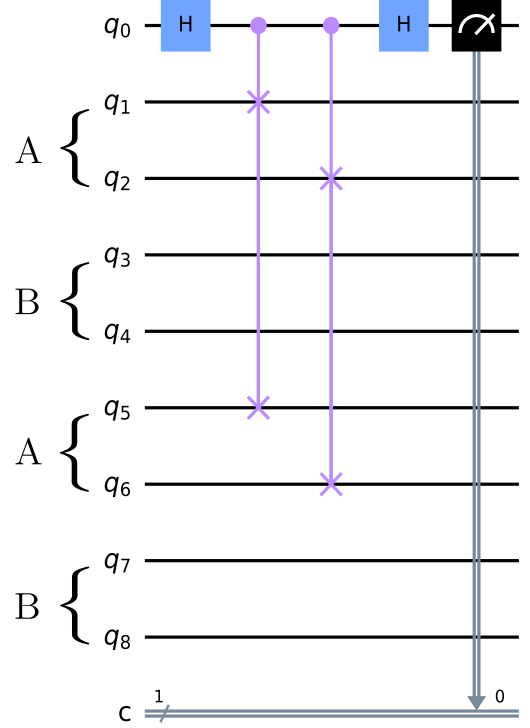


FIG. 5. Quantum circuit used for the estimation of the reduced purity  $\gamma_A$  of a system  $AB$  composed by  $q = 4$  qubits. This circuit is analogous to the general case, with some exceptions. Here, the first  $q$  qubits after the ancilla qubit  $q_0$  and the last  $q$  qubits are prepared in the same state  $\hat{\rho}$ . Also, the controlled-SWAP gates are applied only between the first half of qubits of each copy. Then, measuring  $q_0$  for various identical circuits allows us to estimate the reduced purity  $\gamma_A = \text{Tr}(\hat{\rho}_A^2)$ , which corresponds to the bipartite reduced state of qubits  $q_1$  and  $q_2$  or  $q_5$  and  $q_6$ .

$$\begin{aligned} &= \frac{1}{2}(|+\rangle\langle +| + |-\rangle\langle -| + |-\rangle\langle +| + |+\rangle\langle -|) \\ &\quad \times \text{Tr}[(\hat{\rho}_{S_1} \otimes \hat{\rho}_{S_2})\widehat{\text{SWAP}}] \\ &= \frac{\hat{I} + \text{Tr}(\hat{\rho}_{S_1} \hat{\rho}_{S_2})\hat{Z}}{2}, \end{aligned} \quad (\text{E12})$$

where in the last step we used the identity (E2). Thus the probability  $P_0 = \text{Tr}(|0\rangle\langle 0|\hat{\rho}_0)$  of obtaining state  $|0\rangle$  for the ancilla qubit is

$$\begin{aligned} P_0 &= \text{Tr}\left(|0\rangle\langle 0|\left(\frac{\hat{I} + \text{Tr}(\hat{\rho}_{S_1} \hat{\rho}_{S_2})\hat{Z}}{2}\right)\right) \\ &= \frac{1}{2}\text{Tr}(|0\rangle\langle 0| + \text{Tr}(\hat{\rho}_{S_1} \hat{\rho}_{S_2})|0\rangle\langle 0|) \\ &= \frac{1 + \text{Tr}(\hat{\rho}_{S_1} \hat{\rho}_{S_2})}{2}. \end{aligned} \quad (\text{E13})$$

Therefore, we find that  $\text{Tr}(\hat{\rho}_{S_1} \hat{\rho}_{S_2}) = 2P_0 - 1$ . Now, specializing to the case where  $\hat{\rho}_{S_1} = \hat{\rho}_{S_2}$ , we obtain the purity

$$\gamma_{S_1} = \text{Tr}(\hat{\rho}_{S_1}^2) = 2P_0 - 1. \quad (\text{E14})$$

In our quantum algorithm,  $\gamma_{S_1} = \gamma_A$  is a function of time and is actually the reduced purity of subsystem  $A$ . The quantum circuit for this special case is shown in Fig. 5 for two identical systems  $A$  and  $B$  with  $q = 4$  qubits each.



- [1] D. M. Bressoud, *Factorization and Primality Testing* (Springer-Verlag, New York, 1989).
- [2] R. Crandall and C. Pomerance, *Prime Numbers: A Computational Perspective* (Springer, New York, 2005).
- [3] R. Baillie, A. Fiori, and S. S. Wagstaff, Jr., Strengthening the Baillie-PSW primality test, *Math. Comp.* **90**, 1931 (2021).
- [4] A. Granville, It is easy to determine whether a given integer is prime, *Bull. Amer. Math. Soc.* **42**, 3 (2005).
- [5] D. Schumayer and D. A. W. Hutchinson, Colloquium: Physics of the Riemann hypothesis, *Rev. Mod. Phys.* **83**, 307 (2011).
- [6] M. Wolf, Will a physicist prove the Riemann hypothesis? *Rep. Prog. Phys.* **83**, 036001 (2020).
- [7] C. Feiler and W. P. Schleich, Entanglement and analytical continuation: An intimate relation told by the Riemann zeta function, *New J. Phys.* **15**, 063009 (2013).
- [8] G. Sierra and P. K. Townsend, Landau levels and Riemann Zeros, *Phys. Rev. Lett.* **101**, 110201 (2008).
- [9] S. Aaronson, The Prime Facts: From Euclid to AKS, <https://www.scottaaronson.com/writings/prime.pdf>.
- [10] M. Agrawal, N. Kayal, and N. Saxena, PRIMES is in P, *Ann. Math.* **160**, 781 (2004).
- [11] H. A. Helfgott, An improved sieve of Eratosthenes, *Math. Comp.* **89**, 333 (2020).
- [12] G. L. Miller, Riemann's hypothesis and tests for primality, *J. Comput. Syst. Sci.* **13**, 300 (1976).
- [13] L. M. Adleman, On distinguishing prime numbers from composite numbers, *Ann. Math.* **117**, 173 (1983).
- [14] A. Donis-Vela and J. C. Garcia-Escartin, A quantum primality test with order finding, *Quantum Inf. Comput.* **18**, 1143 (2018).
- [15] H. F. Chau and H.-K. Lo, Primality test via quantum factorization, *Int. J. Mod. Phys. C* **08**, 131 (1997).
- [16] J. Li, X. Peng, J. Du, and D. Suter, An efficient exact quantum algorithm for the integer square-free decomposition problem, *Sci. Rep.* **2**, 260 (2012).
- [17] D. García-Martín, E. Ribas, S. Carrazza, J. I. Latorre, and G. Sierra, The Prime state and its quantum relatives, *Quantum* **4**, 371 (2020).
- [18] G. Mussardo, A. Trombettoni, and Z. Zhang, Prime suspects in a quantum ladder, *Phys. Rev. Lett.* **125**, 240603 (2020).
- [19] A. L. M. Southier, L. F. Santos, P. H. S. Ribeiro, and A. D. Ribeiro, Identifying primes from entanglement dynamics, *Phys. Rev. A* **108**, 042404 (2023).
- [20] S. S. Bullock and I. L. Markov, Asymptotically optimal circuits for arbitrary  $n$ -qubit diagonal computations, *Quantum Inf. Comput.* **4**, 27 (2004).
- [21] J. Welch, D. Greenbaum, S. Mostame, and A. Aspuru-Guzik, Efficient quantum circuits for diagonal unitaries without ancillas, *New J. Phys.* **16**, 033040 (2014).
- [22] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher, Concentrating partial entanglement by local operations, *Phys. Rev. A* **53**, 2046 (1996).
- [23] G. Vidal and R. Tarrach, Robustness of entanglement, *Phys. Rev. A* **59**, 141 (1999).
- [24] M. L. W. Basso and J. Maziero, Entanglement monotones from complementarity relations, *J. Phys. A: Math. Theor.* **55**, 355304 (2022).
- [25] M. V. Scherer and A. D. Ribeiro, Entanglement dynamics of spins using a few complex trajectories, *Phys. Rev. A* **104**, 042222 (2021).
- [26] A. K. Ekert, C. M. Alves, D. K. L. Oi, M. Horodecki, P. Horodecki, and L. C. Kwek, Direct estimations of linear and nonlinear functionals of a quantum state, *Phys. Rev. Lett.* **88**, 217901 (2002).
- [27] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing* (Cambridge University Press, New York, 2007).
- [28] A. Javadi-Abhari *et al.*, Quantum computing with Qiskit, [arXiv:2405.08810](https://arxiv.org/abs/2405.08810).
- [29] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, UK, 2000).
- [30] J. Maziero, Computing partial traces and reduced density matrices, *Int. J. Mod. Phys. C* **28**, 1750005 (2017).
- [31] G. B. Arfken, H. J. Weber, and F. E. Harris, *Mathematical Methods for Physicists: A Comprehensive Guide*, 7th ed. (Elsevier, Oxford, 2013).
- [32] J. L. Walsh, A closed set of normal orthogonal functions, *Am. J. Math.* **45**, 5 (1923).
- [33] N. J. Fine, On the Walsh functions, *Trans. Am. Math. Soc.* **65**, 372 (1949).
- [34] L. Zhihua and Z. Qishan, Ordering of Walsh functions, *IEEE Trans. Electromagn. Compat.* **EMC-25**, 115 (1983).
- [35] C.-K. Yuen, Function approximation by Walsh series, *IEEE Trans. Comput.* **C-24**, 590 (1975).
- [36] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf, Quantum fingerprinting, *Phys. Rev. Lett.* **87**, 167902 (2001).
- [37] A. Barenco, A. Berthiaume, D. Deutsch, A. Ekert, R. Jozsa, and C. Macchiavello, Stabilization of quantum computations by symmetrization, *SIAM J. Comput.* **26**, 1541 (1997).
- [38] <https://github.com/santosvictorf/primers-identification-using-qcomputers/tree/main/qiskit>.