

Quantum Algorithms for Matrix Operations of Row Addition, Row Swapping, Trace Calculation and Transpose

Yu-Hang Liu¹, Yuan-Hong Tao^{1,*}, Jing-Run Lan¹ and Shao-Ming Fei²

¹ School of Science, Zhejiang University of Science and Technology, Hangzhou 310023, China

² School of Mathematical Sciences, Capital Normal University, Beijing 100048, China

* Corresponding author: taoyuanhong12@126.com

Received date: 17 July 2025; Accepted date: 10 September 2025; Published online: 28 November 2025

Abstract: Quantum algorithms of matrix operations are of great significance in many fields in science and technology. In this paper, by leveraging multi-qubit Toffoli gates and basic single-qubit operations, the quantum algorithms of matrix operations of row addition, row swapping, trace calculation and transpose are obtained. In particular, the complexities of these quantum algorithms are presented, too.

Keywords: quantum algorithm; quantum circuit; matrix operation; row addition; row swapping; quantum measurement

1. Introduction

Quantum computations offer exponential speedups over their classical counterparts by leveraging the principles of quantum mechanics. In 1980 Benioff and Manin [1] proposed the concept of quantum computing, and Feynman [2] introduced the theoretical framework of a quantum computer in 1982. In 1985 Deutsch [3] proposed the quantum parallel algorithm, showcasing the significant computational advantages of quantum computing over classical methods.

In 1994 Shor [4,5] presented a quantum algorithm for prime factorization, drawing widespread attention to the advantages of quantum computing in solving certain problems. Grover [6] followed in 1996 with a quantum search algorithm, further demonstrating quantum computing's potential in data search applications. In 2009 the HHL algorithm [7,8] was developed for solving linear systems of equations, greatly advancing the quantum algorithm researches. Several renowned quantum operations, such as quantum Fourier transform [9–11] and quantum phase estimation [11,12], have significantly contributed to the field.

Matrix operations serve as the core computational power support for quantum algorithms. The efficient quantum implementation of its basic operations (including row transformation, transposition, and trace calculation, etc.) holds significant theoretical and practical value. In quantum chemistry simulations [13], the elementary row transformation of matrices can rapidly simplify the Hamiltonian matrix, reducing the time complexity of solving molecular energy spectra from the polynomial level of classical algorithms to the logarithmic level. Matrix Transpose operation enables the efficient analysis of molecular symmetry, while the calculation of the matrix trace can directly yield key physical quantities such as the number of electrons. In quantum cryptography [14], elementary row transformation can optimize the encoding of quantum states to enhance security. Matrix Transpose can achieve efficient information routing in quantum communication networks, and the matrix trace can be used for the integrity detection of quantum states. In the field of quantum machine learning [15], elementary row transformation can accelerate the feature extraction of Quantum Support Vector Machines (QSVMs). Matrix Transpose can improve the backpropagation efficiency of quantum neural networks, and the matrix trace operation can quickly calculate kernel functions to evaluate model performance. Given the wide-ranging influence of matrix operations in interdisciplinary fields, exploring efficient quantum computing methods to solve matrix-related algebraic problems has become a current research hotspot.

In recent years, researchers have proposed several quantum algorithms for matrix operations. For example, the Trotter approximation formula [16] has been utilized to simulate quantum circuits, addressing complex operations such as matrix addition, multiplication and Kronecker products. Quantum algorithms for computing vector inner products [17–22] have also been developed. These algorithms leverage the superposition and entanglement of qubits and have potential applications in fields like machine learning and data processing. Based on a “sender-receiver” quantum computing model, in 2022 Wen-Tao Qi et al. [19] proposed a class of quantum algorithms for matrix operations, including matrix-vector multiplication, matrix-matrix multiplication, matrix addition, determinant computation and matrix inversion. In 2023 A.I. Zenchuk et al. [20] further developed the idea of using specific types of unitary transformations to implement linear algebra protocols according to multi-qubit Toffoli gates and the simplest single-qubit operations. Auxiliary measurements were employed to remove redundant information and extract desired results. Building on this work, in 2024 they [21] also introduced a method to encode the N rows of a matrix into pure states of N independent quantum subsystems. By selecting appropriate unitary operators, the determinant can be computed efficiently. Based on this, they [22] proposed a variational quantum singular value decomposition method based on matrix encoding, which reduced the number of measurements through controlled measurement. Furthermore, in 2025 they [23] discussed the method of encoding unnormalized matrices, and the system encoding the matrix was extended adding one more probability amplitude that satisfy the normalization condition.

However, these quantum algorithms for basic matrix operations still face challenges such as high computational complexity and low success probabilities, which limit their broader applicability.

The elementary row transformations of matrices play significant roles in solving linear systems of equations, determining matrix rank, finding invertible matrices and simplifying determinant computations, with wide-ranging value in both theoretical research and practical applications. In quantum computing, elementary row transformations can simplify matrices for efficient linear equation solving, provide theoretical support for matrix decomposition, advance large-scale matrix operations, and help to interpret the geometric meanings of linear transformations. These contributions offer new insights into the designs of quantum algorithms. Currently, there are no specialized quantum algorithms specifically designed for elementary row transformations of matrices.

Drawing on the foundational ideas from [20], this paper proposes algorithms for two elementary row transformations of matrices, transpose and trace of matrices, along with their corresponding quantum circuits. These algorithms collectively involve the following key steps:

1. Construction of the initial state: matrix elements are first encoded into the probability amplitudes of quantum pure states by using tensor product operations.
2. Information separation and labeling: combining multi-qubit control operators with ancillary states, where the ancillary states are used to accurately label the useful and redundant parts of the information, achieving effective separation of the data.
3. State swapping operations: depending on the specific algorithms, C-SWAP operations are flexibly employed to precisely exchange the states of two quantum bits.
4. State superposition processing: Hadamard operators are applied to induce superposition states among quantum bits.
5. Measurement on the ancillary states: auxiliary measurement techniques are used to eliminate redundant information, ensuring that the desired results are accurately reflected in the probability amplitudes of the final state.

This paper focuses on the core algorithm layer of quantum computers and aims to design a set of quantum circuit schemes for implementing basic matrix operations. The research is carried out under the following premise: the input state, which is the quantum state encoding matrix elements, is successfully prepared. On this basis, we systematically construct a quantum circuit architecture capable of performing basic matrix operations such as row transformation transposition, and trace calculation.

This approach is of great value in the current development of quantum computing. On the one hand, quantum state preparation and quantum state measurement, as crucial aspects in quantum computing, have attracted much attention on conducting specialized studies. Transforming classical matrices into quantum pure states is a fundamental issue that needs to be addressed in the field of quantum state preparation. Moreover, converting the results of quantum state measurements into classical data interpretable by humans makes quantum readout the core part of quantum

state measurement research. On the other hand, the algorithm designed in this paper fully considers the compatibility with quantum state preparation and quantum state measurement schemes, laying a solid foundation for the rapid deployment and application of quantum computers after they become practically implementable in the future.

The remainder of this paper is organized as follows: Section 2 introduces quantum algorithms for two types of elementary matrix transformations. Section 3 presents quantum algorithms for computing the trace and the transposition of a matrix. Conclusion is given in section 2.

2. Two Types of Elementary Row Transformations

This section presents quantum algorithms for two types of elementary row transformations of an $N \times M$ matrix $A = \{a_{ij}\}$ ($N = 2^n$, $M = 2^m$), including adding one row to another and swapping two rows.

2.1. Adding One Row of a Matrix to Another Row

For the quantum algorithm that adds one row of a matrix to another, the quantum circuit diagram is shown in Figure 1. This algorithm requires two pure states and four auxiliary states, and is divided into seven steps.

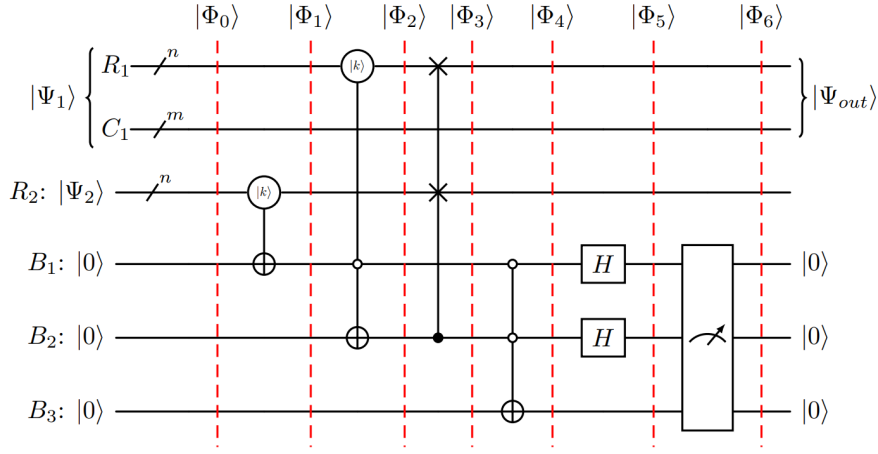


Figure 1. Quantum circuit diagram for adding one row of a matrix to another row.

The following steps give rise to the algorithm of adding the $(k + 1)$ th row of a matrix A to the $(l + 1)$ th row with $0 \leq l, k \leq N - 1$.

Step 1: Prepare the initial state

First, introduce an n -qubits register R_1 and an m -qubit register C_1 , which are used to enumerate the rows and columns of matrix A , respectively. The entries a_{ij} of the matrix A are then encoded into a pure state as follows,

$$|\Psi_1\rangle = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1}, \quad \sum_{ij} |a_{ij}|^2 = 1,$$

where $|i\rangle$ is the binary representation of i .

Next, introduce an N -dimensional auxiliary column vector Z (with the elements of the $k + 1$ and the $l + 1$ rows being $\frac{1}{\sqrt{2}}$, and all other elements being 0), along with an n -qubit register R_2 . The elements of Z are then encoded into pure states as follows,

$$|\Psi_2\rangle = \frac{1}{\sqrt{2}} (|k\rangle_{R_2} + |l\rangle_{R_2}).$$

The initial state of the entire system is then given by

$$\begin{aligned}
|\Phi_0\rangle &= |\Psi_1\rangle \otimes |\Psi_2\rangle \\
&= \frac{1}{\sqrt{2}} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} \right. \\
&\quad \left. + \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} \right).
\end{aligned} \tag{1}$$

Step 2: Label the two states of R_2 .

In order to accurately separate the useful and useless information terms in $|\Phi_0\rangle$ for subsequent calculations, we need to label the two states of R_2 in $|\Phi_0\rangle$ accordingly.

Introduce a 1-qubit auxiliary B_1 in state $|0\rangle_{B_1}$, and the following control operator,

$$W_{R_2 B_1}^{(1)} = P_{R_2} \otimes \sigma_{B_1}^{(x)} + (I_{R_2} - P_{R_2}) \otimes I_{B_1},$$

where $P_{R_2} = |k\rangle_{R_2} \langle k|$ is a projection operator, $\sigma_{B_1}^{(x)}$ is the standard Pauli matrix, and I_{B_1} is the identity operator acting on B_1 . Applying the operator $W_{R_2 B_1}^{(1)}$ to $|\Phi_0\rangle|0\rangle_{B_1}$, we get

$$\begin{aligned}
|\Phi_1\rangle &= W_{R_2 B_1}^{(1)} |\Phi_0\rangle \otimes |0\rangle_{B_1} \\
&= \frac{1}{\sqrt{2}} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |1\rangle_{B_1} \right. \\
&\quad \left. + \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |0\rangle_{B_1} \right).
\end{aligned} \tag{2}$$

That is, for the term where the state of R_2 is $|k\rangle_{R_2}$, it is labeled by $|1\rangle_{B_1}$, and for the term where the state of R_2 is $|l\rangle_{R_2}$, it is labeled by $|0\rangle_{B_1}$.

Kitaev A. Y. et al. [24] represented an important result: A control operator with n control qubits can be represented by $O(n)$ Toffoli gates. Since the above control operator $W_{R_2 B_1}^{(1)}$ has an n -qubit control register, it can be expressed by using $O(n)$ Toffoli gates. Therefore, the circuit complexity for computing $|\Phi_1\rangle$ is $O(n) = O(\log N)$.

Step 3: Separate the useful information from the redundant information.

To precisely extract the element located at the $(k+1)$ th row of the second term in $|\Phi_1\rangle$, from the terms in $|\Phi_1\rangle$ labeled by $|0\rangle_{B_1}$ we only extract the terms where R_1 is in the state $|k\rangle_{R_1}$. To do so, we introduce a 1-qubit auxiliary B_2 in the state $|0\rangle_{B_2}$, and the following control operator,

$$W_{R_1 B_1 B_2}^{(2)} = P_{R_1 B_1} \otimes \sigma_{B_2}^{(x)} + (I_{R_1 B_1} - P_{R_1 B_1}) \otimes I_{B_2},$$

where $P_{R_1 B_1} = |k\rangle_{R_1} \langle k|_{B_1}$ is a projection operator.

Applying the operator $W_{R_1 B_1 B_2}^{(2)}$ to $|\Phi_1\rangle|0\rangle_{B_2}$, we obtain

$$\begin{aligned}
|\Phi_2\rangle &= W_{R_1 B_1 B_2}^{(2)} |\Phi_1\rangle \otimes |0\rangle_{B_2} \\
&= \frac{1}{\sqrt{2}} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |1\rangle_{B_1} |0\rangle_{B_2} \right. \\
&\quad \left. + \sum_{j=0}^{M-1} a_{kj} |k\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |0\rangle_{B_1} |1\rangle_{B_2} \right) \\
&\quad + |g_1\rangle_{R_1 C_1 R_2} |0\rangle_{B_1} |0\rangle_{B_2}.
\end{aligned} \tag{3}$$

That is, the terms where the state of R_1 and B_1 are $|k\rangle_{R_1} |0\rangle_{B_1}$ are labeled by $|1\rangle_{B_2}$, while the other terms are labeled by $|0\rangle_{B_2}$. The states $|1\rangle_{B_1} |0\rangle_{B_2}$ and $|0\rangle_{B_1} |1\rangle_{B_2}$ in $|\Phi_2\rangle$ correspond to the useful information terms required by the algorithm. All other useless information terms are represented by $|g_1\rangle_{R_1 C_1 R_2}$.

The control operator $W_{R_1 B_1 B_2}^{(2)}$ has $n + 1$ control qubits, so it can be represented in terms of $O(n)$ Toffoli gates and the circuit complexity for computing $|\Phi_2\rangle$ is $O(n) = O(\log N)$.

Step 4: Use C-SWAP gate to exchange the states of R_1 and R_2 marked by $|1\rangle_{B_2}$.

To add the $(k + 1)$ th row to the $(l + 1)$ th row, we apply a C-SWAP gate operation to the terms in $|\Phi_2\rangle$ labeled by $|1\rangle_{B_2}$, which swaps the states of R_1 and R_2 . This operation is given by the following gate,

$$W_{R_1 R_2 B_2}^{(3)} = \text{SWAP}_{R_1, R_2} \otimes |1\rangle_{B_2} \langle 1| + I_{R_1 R_2} \otimes |0\rangle_{B_2} \langle 0|.$$

Applying $W_{R_1 R_2 B_2}^{(3)}$ to $|\Phi_2\rangle$ we obtain

$$\begin{aligned} |\Phi_3\rangle &= W_{R_1 R_2 B_2}^{(3)} |\Phi_2\rangle \\ &= \frac{1}{\sqrt{2}} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |1\rangle_{B_1} |0\rangle_{B_2} \right. \\ &\quad \left. + \sum_{j=0}^{M-1} a_{kj} |l\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |0\rangle_{B_1} |1\rangle_{B_2} \right) \\ &\quad + |g_1\rangle_{R_1 C_1 R_2} |0\rangle_{B_1} |0\rangle_{B_2}. \end{aligned} \quad (4)$$

Note that the SWAPs in the control operator $W_{R_1 R_2 B_2}^{(3)}$ have common single control qubit and are related to different pairs of qubits. Therefore, they can be applied simultaneously.

Step 5: Label the useful and redundant information.

To effectively avoid mixing useful and useless information terms in the subsequent computation process, we introduce an auxiliary qubit B_3 in state $|0\rangle_{B_3}$, together with the projection operator $P_{B_1 B_2} = |00\rangle_{B_1 B_2} \langle 00|$. We use B_3 to label the useful and useless information terms in $|\Phi_3\rangle$ accordingly by using the following control operator,

$$W_{B_1 B_2 B_3}^{(4)} = P_{B_1 B_2} \otimes \sigma_{B_3}^{(x)} + (I_{B_1 B_2} - P_{B_1 B_2}) \otimes I_{B_3}.$$

Applying $W_{B_1 B_2 B_3}^{(4)}$ to $|\Phi_3\rangle |0\rangle_{B_3}$ we obtain

$$\begin{aligned} |\Phi_4\rangle &= W_{B_1 B_2 B_3}^{(4)} |\Phi_3\rangle |0\rangle_{B_3} \\ &= \frac{1}{\sqrt{2}} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |1\rangle_{B_1} |0\rangle_{B_2} \right. \\ &\quad \left. + \sum_{j=0}^{M-1} a_{kj} |l\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |0\rangle_{B_1} |1\rangle_{B_2} \right) |0\rangle_{B_3} \\ &\quad + |g_1\rangle_{R_1 C_1 R_2 B_1 B_2} |1\rangle_{B_3}. \end{aligned} \quad (5)$$

In this way, $|0\rangle_{B_3}$ ($|1\rangle_{B_3}$) labels the useful (useless) information terms.

Since the control operator $W_{B_1 B_2 B_3}^{(4)}$ has two control qubit, its computational complexity is $O(1)$.

Step 6: Use a Hadamard gate for information superposition.

Applying the Hadamard operator $W_{B_1 B_2}^{(5)} = H^{\otimes 2}$ to B_1 and B_2 , we obtain

$$\begin{aligned} |\Phi_5\rangle &= W_{B_1 B_2}^{(5)} |\Phi_4\rangle \\ &= \frac{1}{(\sqrt{2})^3} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} \right. \\ &\quad \left. + \sum_{j=0}^{M-1} a_{kj} |l\rangle_{R_1} |j\rangle_{C_1} \right) |k\rangle_{R_2} |0\rangle_{B_1} |0\rangle_{B_2} |0\rangle_{B_3} \\ &\quad + |g_2\rangle_{R_1 C_1 R_2 B_1 B_2 B_3}. \end{aligned} \quad (6)$$

Here, the terms in the state $|0\rangle_{B_1}|0\rangle_{B_2}|0\rangle_{B_3}$ are retained, which contain all the information required for further process. The remaining terms are represented as $|\mathcal{G}_2\rangle_{R_1C_1R_2B_1B_2B_3}$.

Since only two Hadamard gates are used here, the circuit complexity is $O(1)$.

Step 7: Measure the auxiliary state $B_1B_2B_3$ using $|000\rangle_{B_1B_2B_3}$.

By measuring the auxiliary $B_1B_2B_3$ in the state $|000\rangle_{B_1B_2B_3}$, we get

$$|\Phi_6\rangle = |\Psi_{out}\rangle|k\rangle_{R_2}, \quad (7)$$

where

$$|\Psi_{out}\rangle = \frac{1}{G} \left(\sum_{i \neq l} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} + \sum_{j=0}^{M-1} (a_{lj} + a_{kj}) |l\rangle_{R_1} |j\rangle_{C_1} \right) \quad (8)$$

with the normalization

$$G = \left(\sum_{i \neq l} \sum_j |a_{ij}|^2 + \sum_j |a_{kj} + a_{lj}|^2 \right)^{1/2}.$$

The success probability is $\frac{G^2}{8}$. This probability clearly depends only on the matrix elements a_{ij} and is independent of the matrix dimension.

The overall computational complexity of the algorithm is $O(n)$, mainly determined by $W_{R_2B_1}^{(1)}$ and $W_{R_1B_1B_2}^{(2)}$.

2.2. Swapping Two Rows of a Matrix

For the quantum algorithm that swaps two rows of a matrix, the quantum circuit diagram is shown in Figure 2. The algorithm requires two pure states and four auxiliaries, and is divided into seven steps to swap the $(k+1)$ th and $(l+1)$ th rows of matrix A ($0 \leq l, k \leq N-1$).

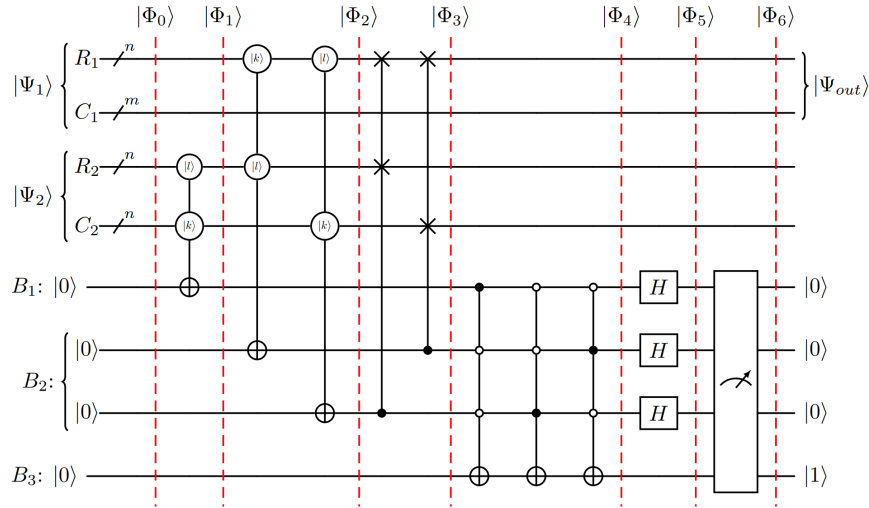


Figure 2. The quantum circuit diagram for swapping two rows of a matrix.

Step 1: Prepare the initial state.

First, introduce an n -qubit register R_1 and an m -qubit register C_1 , which enumerate the rows and columns of the matrix A , respectively. The elements of the matrix A are then encoded into the following pure state,

$$|\Psi_1\rangle = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1}, \quad \sum_{ij} |a_{ij}|^2 = 1,$$

Next, introduce an $N \times N$ auxiliary matrix Z such that the elements in the $(l + 1)$ -th row and $(k + 1)$ -th column, the $(k + 1)$ -th row and $(k + 1)$ -th column, and the $(l + 1)$ -th row and $(l + 1)$ -th column are $\frac{1}{\sqrt{3}}$, the rest elements are 0; along with two n -qubit registers R_2 and C_2 . The elements of Z are then encoded into the following pure state,

$$|\Psi_2\rangle = \frac{1}{\sqrt{3}}(|l\rangle_{R_2}|k\rangle_{C_2} + |k\rangle_{R_2}|k\rangle_{C_2} + |l\rangle_{R_2}|l\rangle_{C_2}).$$

The initial state is of the form,

$$\begin{aligned} |\Phi_0\rangle &= |\Psi_1\rangle \otimes |\Psi_2\rangle \\ &= \frac{1}{\sqrt{3}} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} \right. \\ &\quad + \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |k\rangle_{C_2} \\ &\quad \left. + \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |l\rangle_{C_2} \right). \end{aligned} \quad (9)$$

Step 2: Label the distinct states of R_2 and C_2 .

To accurately separate the useful from the useless information terms in $|\Phi_0\rangle$, introduce an auxiliary qubit B_1 in state $|0\rangle_{B_1}$, and the following control operator,

$$W_{R_2 C_2 B_1}^{(1)} = P_{R_2 C_2} \otimes \sigma_{B_1}^{(x)} + (I_{R_2 C_2} - P_{R_2 C_2}) \otimes I_{B_1},$$

where $P_{R_2 C_2} = |l\rangle_{R_2} |k\rangle_{C_2} \langle l|_{R_2} \langle k|_{C_2}$ is the projection operator. Applying $W_{R_2 C_2 B_1}^{(1)}$ to $|\Phi_0\rangle |0\rangle_{B_1}$, we get

$$\begin{aligned} |\Phi_1\rangle &= W_{R_2 C_2 B_1}^{(1)} |\Phi_0\rangle \otimes |0\rangle_{B_1} \\ &= \frac{1}{\sqrt{3}} \left(\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |1\rangle_{B_1} \right. \\ &\quad + \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |k\rangle_{C_2} |0\rangle_{B_1} \\ &\quad \left. + \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |l\rangle_{C_2} |0\rangle_{B_1} \right). \end{aligned} \quad (10)$$

In this way, the terms where the states of R_2 and C_2 are different are labeled by $|1\rangle_{B_1}$, and the terms where the states of R_2 and C_2 are the same are labeled by $|0\rangle_{B_1}$.

Since the control operator $W_{R_2 C_2 B_1}^{(1)}$ has n control qubit, its computational complexity is $O(n) = O(\log N)$.

Step 3: Separate the useful information terms from the useless information terms.

We proceed to separate the useful information terms from the useless ones.

- 1) In the first sum, extract all elements except for those in the $(k + 1)$ th and $(l + 1)$ th rows.
- 2) In the second sum, extract all elements in the $(l + 1)$ th row.
- 3) In the third sum, extract all elements in the $(k + 1)$ th row.
- 4) For the terms in $|\Phi_1\rangle$ labeled by $|1\rangle_{B_1}$, extract all terms except those where R_1 is in the states $|k\rangle_{R_1}$ and $|l\rangle_{R_1}$.
- 5) For the two sums in $|\Phi_1\rangle$ labeled by $|0\rangle_{B_1}$, extract the terms where R_1 is in the state $|l\rangle_{R_1}$ and the terms where R_1 is in the state $|k\rangle_{R_1}$.

We introduce an auxiliary qubit B_2 in state $|0\rangle_{B_2}$, and use the following control operators,

$$\begin{aligned} W^{(1)} &= P_{R_1 R_2} \otimes \sigma_{B_2}^{(x)} + (I_{R_1 R_2} - P_{R_1 R_2}) \otimes I_{B_2}, \\ W^{(2)} &= P_{R_1 C_2} \otimes \sigma_{B_2}^{(x)} + (I_{R_1 C_2} - P_{R_1 C_2}) \otimes I_{B_2}, \end{aligned}$$

where $P_{R_1 R_2} = |k\rangle_{R_1} \langle l|_{R_2}$, $P_{R_1} = |k\rangle_{R_2} \langle l|$ and $P_{R_1 C_2} = |l\rangle_{R_1} \langle k|_{C_2}$, $P_{R_1} = |l\rangle_{R_2} \langle k|_{C_2}$ are projection operators. $\sigma_{B_2^i}^{(x)}$ and $I_{B_2^i}$ are the operators acting on the i -th qubit of the auxiliary B_2 ($i = 1, 2$).

When $W_{R_1 R_2 C_2 B_2}^{(2)} = W^{(2)} W^{(1)}$ acts on $|\Phi_1\rangle|0\rangle_{B_2}$, we obtain

$$\begin{aligned} |\Phi_2\rangle &= W_{R_1 R_2 C_2 B_2}^{(2)} |\Phi_1\rangle \otimes |0\rangle_{B_2} \\ &= \frac{1}{\sqrt{3}} \left(\sum_{i \neq l, k} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |1\rangle_{B_1} |00\rangle_{B_2} \right. \\ &\quad + \sum_{j=0}^{M-1} a_{lj} |l\rangle_{R_1} |j\rangle_{C_1} |k\rangle_{R_2} |k\rangle_{C_2} |0\rangle_{B_1} |01\rangle_{B_2} \\ &\quad + \left. \sum_{j=0}^{M-1} a_{kj} |k\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |l\rangle_{C_2} |0\rangle_{B_1} |10\rangle_{B_2} \right) \\ &\quad + |g_1\rangle_{R_1 C_1 R_2 C_2 B_1 B_2}. \end{aligned} \quad (11)$$

That is, the terms labeled by $|1\rangle_{B_1} |00\rangle_{B_2}$, $|0\rangle_{B_1} |01\rangle_{B_2}$ and $|0\rangle_{B_1} |10\rangle_{B_2}$ are the useful information terms required by the algorithm, while the other terms are the useless information terms denoted by $|g_1\rangle_{R_1 C_1 R_2 C_2 B_1 B_2}$.

Since the control operator $W_{R_1 R_2 C_2 B_2}^{(2)}$ has $2n$ control qubit, its computational complexity is $O(n) = O(\log N)$.

Step 4: Use C-SWAP gates to swap the states of R_1 and R_2 , as well as the states of R_1 and C_2 .

To swap the $(k+1)$ th and $(l+1)$ th rows of the matrix, we apply C-SWAP gate to the terms in $|\Phi_2\rangle$ labeled by $|10\rangle_{B_2}$ and $|01\rangle_{B_2}$, so as to swap the states of R_1 and R_2 , as well as the states of R_1 and C_2 . Denote

$$\begin{aligned} W^{(1)} &= SWAP_{R_1, C_2} \otimes |1\rangle_{B_2^1} \langle 1| + I_{R_1 C_2} \otimes |0\rangle_{B_2^1} \langle 0|, \\ W^{(2)} &= SWAP_{R_1, R_2} \otimes |1\rangle_{B_2^2} \langle 1| + I_{R_1 R_2} \otimes |0\rangle_{B_2^2} \langle 0|, \end{aligned}$$

where $|1\rangle_{B_2^1}$ stands for that the first qubit of B_2 is in the state $|1\rangle$, and $|1\rangle_{B_2^2}$ for that the second qubit of B_2 in the state $|1\rangle$.

Applying $W_{R_1 R_2 C_2 B_2}^{(3)} = W^{(2)} W^{(1)}$ to $|\Phi_2\rangle$, we obtain

$$\begin{aligned} |\Phi_3\rangle &= W_{R_1 R_2 C_2 B_2}^{(3)} |\Phi_2\rangle \\ &= \frac{1}{\sqrt{3}} \left(\sum_{i \neq l, k} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |1\rangle_{B_1} |00\rangle_{B_2} \right. \\ &\quad + \sum_{j=0}^{M-1} a_{lj} |k\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |0\rangle_{B_1} |01\rangle_{B_2} \\ &\quad + \left. \sum_{j=0}^{M-1} a_{kj} |l\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |0\rangle_{B_1} |10\rangle_{B_2} \right) + \\ &\quad + |g_2\rangle_{R_1 C_1 R_2 C_2 B_1 B_2}. \end{aligned} \quad (12)$$

The SWAP gates transfer the elements originally in the $(k+1)$ th row to the $(l+1)$ th row in the terms labeled by $|0\rangle_{B_1} |01\rangle_{B_2}$, and the elements originally in the $(l+1)$ th row to the $(k+1)$ th row in the terms labeled by $|0\rangle_{B_1} |10\rangle_{B_2}$.

Note that the SWAPs in the control operator $W_{R_1 R_2 B_2}^{(3)}$ have common single control qubit and are related to different pairs of qubits. Therefore, they can be applied simultaneously.

Step 5: Label the useful and useless information terms.

Introduce an auxiliary qubit B_3 in state $|0\rangle_{B_3}$. To use B_3 to label the useful and useless information terms in $|\Phi_3\rangle$, we apply the following control operator,

$$W_{B_1 B_2 B_3}^{(4)} = V^{(1)} V^{(2)} V^{(3)},$$

where $V^{(i)} = P_{B_1 B_2}^{(i)} \otimes \sigma_{B_3}^{(x)} + (I_{B_1 B_2} - P_{B_1 B_2}) \otimes I_{B_3}$, with the projection operators $P_{B_1 B_2}^{(i)}$ ($i = 1, 2, 3$) given by

$$\begin{aligned} P_{B_1 B_2}^{(1)} &= |100\rangle_{B_1 B_2} \langle 100|, \\ P_{B_1 B_2}^{(2)} &= |001\rangle_{B_1 B_2} \langle 001|, \\ P_{B_1 B_2}^{(3)} &= |010\rangle_{B_1 B_2} \langle 010|. \end{aligned}$$

Acting $W_{B_1 B_2 B_3}^{(4)}$ on $|\Phi_3\rangle|0\rangle_{B_3}$, we obtain

$$\begin{aligned} |\Phi_4\rangle &= W_{B_1 B_2 B_3}^{(4)} |\Phi_3\rangle|0\rangle_{B_3} \\ &= \frac{1}{\sqrt{3}} \left(\sum_{i \neq l, k} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |1\rangle_{B_1} |00\rangle_{B_2} \right. \\ &\quad + \sum_{j=0}^{M-1} a_{lj} |k\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |0\rangle_{B_1} |01\rangle_{B_2} \\ &\quad \left. + \sum_{j=0}^{M-1} a_{kj} |l\rangle_{R_1} |j\rangle_{C_1} |l\rangle_{R_2} |k\rangle_{C_2} |0\rangle_{B_1} |10\rangle_{B_2} \right) |1\rangle_{B_3} \\ &\quad + |g_2\rangle_{R_1 C_1 R_2 C_2 B_1 B_2} |0\rangle_{B_3}. \end{aligned} \tag{13}$$

Here, $|1\rangle_{B_3}$ labels the useful information term, while $|0\rangle_{B_3}$ labels the useless information term.

Since the control operator $W_{B_1 B_2 B_3}^{(4)}$ has one control qubit, its computational complexity is $O(1)$.

Step 6: Use Hadamard gates for information superposition.

Applying the Hadamard operator $W_{B_1 B_2}^{(5)} = H^{\otimes 3}$ to B_1 and B_2 , we have

$$\begin{aligned} |\Phi_5\rangle &= W_{B_1 B_2}^{(5)} |\Phi_4\rangle \\ &= \frac{1}{(\sqrt{2})^3 \cdot \sqrt{3}} \left(\sum_{i \neq l, k} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} \right. \\ &\quad \left. + \sum_{j=0}^{M-1} a_{lj} |k\rangle_{R_1} |j\rangle_{C_1} + \sum_{j=0}^{M-1} a_{kj} |l\rangle_{R_1} |j\rangle_{C_1} \right) \\ &\quad |l\rangle_{R_2} |k\rangle_{C_2} |0\rangle_{B_1} |00\rangle_{B_2} |1\rangle_{B_3} + |g_3\rangle_{R_1 C_1 R_2 C_2 B_1 B_2 B_3}, \end{aligned} \tag{14}$$

where the terms in the state $|0\rangle_{B_1} |00\rangle_{B_2} |1\rangle_{B_3}$ are retained, the other terms are denoted by $|g_3\rangle_{R_1 C_1 R_2 B_1 B_2 B_3}$.

At this point, we have successfully swapped the elements of the $(k+1)$ -th row of matrix A with the corresponding elements of the $(l+1)$ -th row.

Since only three Hadamard gates are used here, the circuit complexity is $O(1)$.

Step 7: Measure the auxiliary $B_1 B_2 B_3$ in the state $|0001\rangle_{B_1 B_2 B_3}$

We measure the auxiliary $B_1 B_2 B_3$ in the state $|0001\rangle_{B_1 B_2 B_3}$ to remove the useless information terms, and obtain

$$|\Phi_6\rangle = |\Psi_{out}\rangle |l\rangle_{R_2} |k\rangle_{C_2}, \tag{15}$$

where

$$\begin{aligned} |\Psi_{out}\rangle &= \left(\sum_{i \neq l, k} \sum_{j=0}^{M-1} a_{ij} |i\rangle_{R_1} |j\rangle_{C_1} \right. \\ &\quad \left. + \sum_{j=0}^{M-1} a_{lj} |k\rangle_{R_1} |j\rangle_{C_1} + \sum_{j=0}^{M-1} a_{kj} |l\rangle_{R_1} |j\rangle_{C_1} \right). \end{aligned} \tag{16}$$

The success probability is $\frac{1}{24}$. This probability is independent of the matrix dimension.

The overall computational complexity of the algorithm is $O(n)$, mainly determined by $W_{R_2 C_2 B_1}^{(1)}$ and $W_{R_1 R_2 C_2 B_2}^{(2)}$.

For specific examples of this algorithm, please refer to Appendix A.

3. Matrix Trace Calculation and Transpose

In this section we present quantum algorithms for trace calculation of an $N \times N$ matrix and transposition of an $N \times M$ matrix.

3.1. Matrix Trace Calculation

The quantum circuit for calculating the trace of the matrix S is shown in Figure 3. Assume that a matrix $S = \{a_{ik}\}$ is of size $N \times N$ with $N = 2^n$. This algorithm requires one pure state and three auxiliary qubits, and is divided into the following six steps.

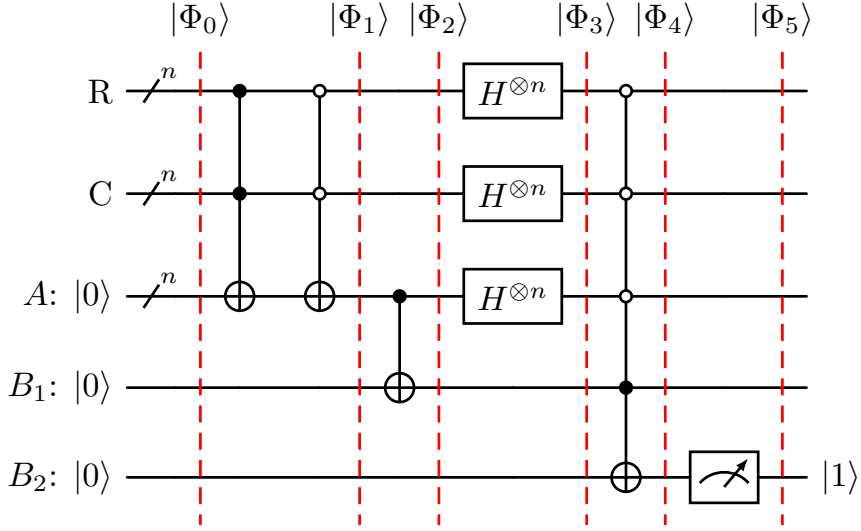


Figure 3. Quantum circuit for calculating matrix trace.

Step 1: Prepare the initial state.

First, introduce two n -qubit registers R and C , which are used to enumerate the rows and columns of the matrix S , and encode the elements of the matrix S to obtain the following pure initial state,

$$|\Phi_0\rangle = \sum_{i=0}^{N-1} \sum_{k=0}^{N-1} a_{ik} |i\rangle_R |k\rangle_C, \quad \sum_{ik} |a_{ik}|^2 = 1. \quad (17)$$

Step 2: Mark the terms where the states of R and C are the same.

To calculate the sum of the diagonal elements of matrix S , we need to mark the diagonal elements of S , i.e., the terms where the states of registers R and C are the same. To do this, introduce an n -qubit auxiliary A in the state $|0\rangle_A$, and an operator

$$W_j^{(m)} = P_j^{(m)} \otimes \sigma_{A^{(j)}}^{(x)} + (I_j - P_j^{(m)}) \otimes I_{A^{(j)}},$$

where $P_j^{(m)} = |m_j\rangle_R |m_j\rangle_C {}_R\langle m_j| {}_C\langle m_j|$, $m = 0, 1, j = 1, 2, \dots, n$, is the projection operator acting on the j -th qubit of registers R and C , $\sigma_{A^{(j)}}^{(x)}$ is the Pauli matrix acting on the j -th qubit of auxiliary A , $I_{A^{(j)}}$ is the identity operator acting on the j -th qubit of auxiliary A .

Applying the operator $W_{RCA}^{(1)} = \prod_{j=1}^n W_j^0 W_j^1$ to $|\Phi_0\rangle |0\rangle_A$, we obtain

$$\begin{aligned} |\Phi_1\rangle &= W_{RCA}^{(1)} |\Phi_0\rangle |0\rangle_A \\ &= \left(\sum_{i=0}^{N-1} a_{ii} |i\rangle_R |i\rangle_C \right) |N-1\rangle_A + |g_1\rangle_{RCA}. \end{aligned} \quad (18)$$

In the process of realizing the quantum state $|\Phi_1\rangle$, we check whether the j -th qubits of registers R and C are the same. If their j -th qubits are the same, the j -th qubit of A is flipped from $|0\rangle$ to $|1\rangle$. If the states of all n qubits of R and C are the same, the states of all n qubits of A are flipped from $|0\rangle_A$ to $|N-1\rangle_A$.

The control operator $W_{RCA}^{(1)}$ has $2n$ control qubits, so the circuit complexity for computing $|\Phi_1\rangle$ is $O(n)$.

Step 3: Mark the useful and useless information terms.

Currently, all useful information terms are stored in the first sum of $|\Phi_1\rangle$, while the second sum $|g_1\rangle_{RCA}$ represents useless information terms that are not needed by the algorithm.

To separately mark the useful and useless information terms in the state $|\Phi_1\rangle$, we introduce the projection operator $P_A = |N-1\rangle_A \langle N-1|$, and a 1-qubit auxiliary B_1 in the state $|0\rangle_{B_1}$. The control operator $W_{AB_1}^{(2)} = P_A \otimes \sigma_{B_1}^{(x)} + (I_A - P_A) \otimes I_{B_1}$ is then applied to A and B_1 , giving rise to

$$\begin{aligned} |\Phi_2\rangle &= W_{AB_1}^{(2)} |\Phi_1\rangle |0\rangle_{B_1} \\ &= \left(\sum_{i=0}^{N-1} a_{ii} |i\rangle_R |i\rangle_C \right) \otimes |N-1\rangle_A \otimes |1\rangle_{B_1} \\ &\quad + |g_1\rangle_{RCA} \otimes |0\rangle_{B_1}. \end{aligned} \quad (19)$$

Here, the useful information terms are marked by $|1\rangle_{B_1}$, and the useless information terms are marked by $|0\rangle_{B_1}$.

Since the control operator $W_{AB_1}^{(2)}$ has n control qubits, its computational complexity is $O(n) = O(\log N)$.

Step 4: Use the Hadamard gate for information superposition.

Up to this point, we have obtained the diagonal elements of S . To compute the sum we apply the Hadamard operator $W_{RCA}^{(3)} = H^{\otimes 3n}$ to all qubits in $|\Phi_2\rangle$ except for the auxiliary B_1 . We get

$$\begin{aligned} |\Phi_3\rangle &= W_{RCA}^{(3)} |\Phi_2\rangle \\ &= \frac{1}{2^{3n/2}} \left(\sum_{i=0}^{N-1} a_{ii} \right) |0\rangle_R |0\rangle_C |0\rangle_A |1\rangle_{B_1} + |g_2\rangle_{RCAB_1}. \end{aligned} \quad (20)$$

In the state $|\Phi_3\rangle$, we select the terms in the state $|0\rangle_R |0\rangle_C |0\rangle_A |1\rangle_{B_1}$, as these terms contain all the useful information required for the trace calculation. The other terms are represented by $|g_2\rangle_{RCAB_1}$.

Since $3n$ Hadamard gates are used here, the circuit complexity is $O(n)$.

Step 5: Re-mark the useful and useless information terms.

Since new useless information terms were generated in the previous step and the useful information is stored in the first term of $|\Phi_3\rangle$, we need to re-mark the useful and useless information terms. To do this, we introduce an auxiliary qubit B_2 in state $|0\rangle_{B_2}$, as well as the projection operator $P_{RCAB_1} = |0\rangle_R |0\rangle_C |0\rangle_A |1\rangle_{B_1} \langle 0|_R \langle 0|_C \langle 0|_A \langle 1|_{B_1}$. We apply the following control operator

$$W_{RCAB_1 B_2}^{(4)} = P_{RCAB_1} \otimes \sigma_{B_2}^{(x)} + (I_{RCAB_1} - P_{RCAB_1}) \otimes I_{B_2}$$

to $|\Phi_3\rangle \otimes |0\rangle_{B_2}$. We obtain

$$\begin{aligned} |\Phi_4\rangle &= W_{RCAB_1 B_2}^{(4)} |\Phi_3\rangle |0\rangle_{B_2} \\ &= \frac{1}{2^{3n/2}} \left(\sum_{i=0}^{N-1} a_{ii} \right) |0\rangle_R |0\rangle_C |0\rangle_A |1\rangle_{B_1} |1\rangle_{B_2} \\ &\quad + |g_2\rangle_{RCAB_1} |0\rangle_{B_2}. \end{aligned} \quad (21)$$

Here, the useful information terms are marked by $|1\rangle_{B_2}$, and the useless information terms are marked by $|0\rangle_{B_2}$.

Since the control operator $W_{RCAB_1 B_2}^{(4)}$ has $3n + 1$ control qubit, its computational complexity is $O(n)$.

Step 6: Measure the auxiliary qubit B_2 in the state $|1\rangle_{B_2}$.

The trace of the matrix S is stored in the quantum state $|\Phi_4\rangle$ in the form of probability amplitudes, with the useful information marked by $|1\rangle_{B_2}$. To extract the useful information, we measure the auxiliary qubit B_2 in the state $|1\rangle_{B_2}$, and obtain

$$|\Phi_5\rangle = \frac{\sum_{i=0}^{N-1} a_{ii}}{|\sum_{i=0}^{N-1} a_{ii}|} |0\rangle_R |0\rangle_C |0\rangle_A |1\rangle_{B_1}. \quad (22)$$

The trace is obtained based on the measurement probability $\frac{|\sum_{i=0}^{N-1} a_{ii}|^2}{2^{3n}}$. This probability is related of the matrix dimension.

The overall computational complexity of the algorithm is $O(n)$, mainly determined by $W_{RCA}^{(1)}$, $W_{AB_1}^{(2)}$, $W_{RCA}^{(3)}$ and $W_{RCAB_1B_2}^{(4)}$.

3.2. Matrix Transpose

The quantum circuit for matrix transpose is shown in Figure 4. Assume that a matrix $S=\{a_{ik}\}$ is of size $N \times M$ with $N = 2^n$ and $M = 2^m$. The algorithm requires two pure states and is divided into two steps.

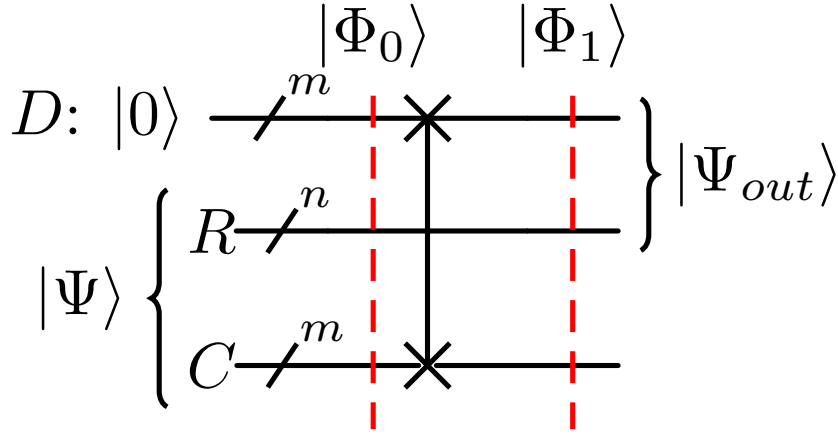


Figure 4. Quantum Circuit for Matrix Transpose.

Step 1: Prepare the initial state.

First, introduce an n -qubit register R and an m -qubit register C , where R and C enumerate the rows and columns of the matrix S , respectively. The elements of matrix S are encoded into the following pure state,

$$|\Psi\rangle = \sum_{i=0}^{N-1} \sum_{k=0}^{M-1} a_{ik} |i\rangle_R |k\rangle_C, \quad \sum_{ik} |a_{ik}|^2 = 1,$$

Then, introduce a pure state $|0\rangle_D$ for an m -qubit register D . The initial state is give by

$$\begin{aligned} |\Phi_0\rangle &= |0\rangle_D \otimes |\Psi\rangle \\ &= \sum_{i=0}^{N-1} \sum_{k=0}^{M-1} a_{ik} |0\rangle_D |i\rangle_R |k\rangle_C. \end{aligned} \quad (23)$$

Step 2: Use a SWAP gate to exchange the states of D and C .

We use a SWAP gate to exchange the states of the D and C registers. The operator $SWAP_{DC}$ is applied to $|\Phi_0\rangle$, yielding

$$\begin{aligned} |\Phi_1\rangle &= SWAP_{DC} |\Phi_0\rangle \\ &= |\Psi_{out}\rangle |0\rangle_C, \end{aligned} \quad (24)$$

where

$$|\Psi_{out}\rangle = \left(\sum_{i=0}^{N-1} \sum_{k=0}^{M-1} a_{ik} |k\rangle_D |i\rangle_R \right). \quad (25)$$

Now, the new rows and new columns of the current target matrix are enumerated by the states D and R respectively. Clearly, the success probability of this algorithm is 1.

Since the exchange operator $SWAP_{DC}$ only involves exchange operations when acting on different qubit pairs, its computational complexity is $O(m)$.

Remark 1. Formally the non-square matrix can be made a square matrix by including the proper number of zero rows or columns. In this case transposition of non-square matrix can be done as a transposition of a square matrix, i.e. by the swap operator $SWAP_{RC}$. The complexity of this operation is $O(n)$.

4. Conclusion

We have developed quantum algorithms for two types of elementary row transformations of matrices, the trace and transpose of matrices. The operational mechanism of all these algorithms is as follows: First, construct the initial state of the entire system. Second, utilize unitary operations and ancillary qubits to mark the information items required by the algorithm. Third, apply the Hadamard transformation to superpose these information items. Finally, measure the ancillary qubits to eliminate the useless information items and retain only the useful ones.

In all these algorithms, the final result is stored in the probability amplitude of certain quantum states. Our success probability for adding one row of a matrix to another is $\frac{G^2}{8}$. The success probability for swapping two rows of a matrix is $\frac{1}{24}$, while the success probability of matrix transpose is 1. The success probability for the trace calculation is $\frac{|\sum_{i=0}^{N-1} a_{ii}|^2}{2^{2m}}$. The success probabilities either depend solely on the elements of the matrix itself, or are fixed or related to the dimensions of the matrix. The larger the dimension, the smaller the success probability. To address this challenge, amplitude amplification techniques may be employed. It is worth noting that the computational complexity of these algorithms is related to the dimension of the matrices. It increases logarithmically as the matrix dimension grows. Specifically, the computational complexity is $O(n)$ for elementary row operations and trace calculation, and $O(m)$ for matrix transpose.

It would be desired to develop new methods to improve the success probability given by the final measurement in the algorithm, or explore the designs of other quantum algorithms for matrix operations based on multi-qubit Toffoli type gates and the simplest single qubit operations. It would be also interesting to conduct further researches on quantum algorithms for other matrix operations.

Author Contributions

All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding

The National Natural Science Foundation of China (Grants No. 11761073, No. 12075159 and No. 12171044), the Academician Innovation Platform of Hainan Province.

Conflicts of Interest Statement

The authors claim that they do not have any Conflict of Interest.

Data Availability Statement

All necessary data are included into the manuscript.

Acknowledgments

We are deeply grateful to Prof. Junde Wu and Alexander I. Zenchuk for their active participation and invaluable contributions to this research.

Appendix A. Swapping Two Rows of a Matrix

For the quantum algorithm that swaps two rows of a matrix, we will provide a detailed explanation of the quantum algorithm that swaps the 4th and 2th rows of the following 4×4 matrix A ,

$$A = \begin{pmatrix} \frac{1}{4} & \frac{1}{16} & \frac{3}{16} & \frac{3}{16} \\ 0 & \frac{1}{2} & \frac{1}{8} & \frac{1}{8} \\ \frac{7}{16} & 0 & \frac{1}{4} & 0 \\ \frac{3}{16} & \frac{3}{16} & \frac{1}{8} & \frac{1}{2} \end{pmatrix}$$

Step 1: Construct the initial state of the entire system.

First, introduce two 2-qubit register R_1 and C_1 , which will enumerate the rows and columns of the matrix A , respectively. The elements of the matrix A are then encoded into the following pure state:

$$\begin{aligned} |\Psi_1\rangle &= \frac{1}{4}|00\rangle_{R_1}|00\rangle_{C_1} + \frac{1}{16}|00\rangle_{R_1}|01\rangle_{C_1} + \frac{3}{16}|00\rangle_{R_1}|10\rangle_{C_1} \\ &+ \frac{3}{16}|00\rangle_{R_1}|11\rangle_{C_1} + 0|01\rangle_{R_1}|00\rangle_{C_1} + \frac{1}{2}|01\rangle_{R_1}|01\rangle_{C_1} \\ &+ \frac{1}{8}|01\rangle_{R_1}|10\rangle_{C_1} + \frac{1}{8}|01\rangle_{R_1}|11\rangle_{C_1} + \frac{7}{16}|10\rangle_{R_1}|00\rangle_{C_1} \\ &+ 0|10\rangle_{R_1}|01\rangle_{C_1} + \frac{1}{4}|10\rangle_{R_1}|10\rangle_{C_1} + 0|10\rangle_{R_1}|11\rangle_{C_1} \\ &+ \frac{3}{16}|11\rangle_{R_1}|00\rangle_{C_1} + \frac{3}{16}|11\rangle_{R_1}|01\rangle_{C_1} + \frac{1}{8}|11\rangle_{R_1}|10\rangle_{C_1} \\ &+ \frac{1}{2}|11\rangle_{R_1}|11\rangle_{C_1}. \end{aligned}$$

Next, introduce an 2×2 auxiliary matrix Z ,

$$Z = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{3}} & 0 & \frac{1}{\sqrt{3}} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{3}} \end{pmatrix},$$

along with two 2-qubit registers R_2 and C_2 . The elements of Z are then encoded into the following pure state:

$$|\Psi_2\rangle = \frac{1}{\sqrt{3}}(|01\rangle_{R_2}|11\rangle_{C_2} + |11\rangle_{R_2}|11\rangle_{C_2} + |01\rangle_{R_2}|01\rangle_{C_2}).$$

The two pure states $|\Psi_1\rangle$ and $|\Psi_2\rangle$ can be tensor-producted to construct the initial state of the entire system

$$\begin{aligned} |\Phi_0\rangle &= |\Psi_1\rangle \otimes |\Psi_2\rangle \\ &= \frac{1}{\sqrt{3}}(|\Psi_1\rangle|01\rangle_{R_2}|11\rangle_{C_2} + |\Psi_1\rangle|11\rangle_{R_2}|11\rangle_{C_2} \\ &\quad + |\Psi_1\rangle|01\rangle_{R_2}|01\rangle_{C_2}) \end{aligned} \tag{A1}$$

Step 2: Label the distinct states of R_2 and C_2 .

Introduce a 1-qubit auxiliary B_1 with the state $|0\rangle_{B_1}$, and construct the following control operator:

$$W_{R_2C_2B_1}^{(1)} = P_{R_2C_2} \otimes \sigma_{B_1}^{(x)} + (I_{R_2C_2} - P_{R_2C_2}) \otimes I_{B_1},$$

where $P_{R_2 C_2} = |01\rangle_{R_2} \langle 11|_{C_2} R_2 \langle 01|_{C_2} \langle 11|$ is the projection operator. When the above operator acts on $|\Phi_0\rangle|0\rangle_{B_1}$, we get:

$$\begin{aligned} |\Phi_1\rangle &= W_{R_2 C_2 B_1}^{(1)} |\Phi_0\rangle \otimes |0\rangle_{B_1} \\ &= \frac{1}{\sqrt{3}} (|\Psi_1\rangle|01\rangle_{R_2} |11\rangle_{C_2} |1\rangle_{B_1} \\ &\quad + |\Psi_1\rangle|01\rangle_{R_2} |01\rangle_{C_2} |0\rangle_{B_1} + |\Psi_1\rangle|11\rangle_{R_2} |11\rangle_{C_2} |0\rangle_{B_1}). \end{aligned} \quad (\text{A2})$$

That is, label the terms where the states of R_2 and C_2 are different with $|1\rangle_{B_1}$, and label the terms where the states of R_2 and C_2 are the same with $|0\rangle_{B_1}$.

Step 3: Separate the useful information terms from the useless information terms.

Introducing a 1-qubit ancilla B_2 with a state of $|0\rangle_{B_2}$, construct the following control operator

$$\begin{aligned} W^{(1)} &= P_{R_1 R_2} \otimes \sigma_{B_2}^{(x)} + (I_{R_1 R_2} - P_{R_1 R_2}) \otimes I_{B_2}, \\ W^{(2)} &= P_{R_1 C_2} \otimes \sigma_{B_2}^{(x)} + (I_{R_1 C_2} - P_{R_1 C_2}) \otimes I_{B_2}, \end{aligned}$$

where $P_{R_1 R_2} = |11\rangle_{R_1} \langle 01|_{R_2} R_1 \langle 11|_{R_2} \langle 01|$, $P_{R_1 C_2} = |01\rangle_{R_1} \langle 11|_{C_2} R_1 \langle 01|_{C_2} \langle 11|$ is the projection operator. Apply $W_{R_1 R_2 C_2 B_2}^{(2)} = W^{(2)} W^{(1)}$ to $|\Phi_1\rangle|0\rangle_{B_2}$, obtained:

$$\begin{aligned} |\Phi_2\rangle &= W_{R_1 R_2 C_2 B_2}^{(2)} |\Phi_1\rangle |0\rangle_{B_2} \\ &= \frac{1}{\sqrt{3}} \left[\left(\frac{1}{4} |00\rangle_{R_1} |00\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \right. \right. \\ &\quad + \frac{1}{16} |00\rangle_{R_1} |01\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \\ &\quad + \frac{3}{16} |00\rangle_{R_1} |10\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \\ &\quad + \frac{3}{16} |00\rangle_{R_1} |11\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \\ &\quad + \frac{7}{16} |10\rangle_{R_1} |00\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \\ &\quad + 0 |10\rangle_{R_1} |01\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \\ &\quad + \frac{1}{4} |10\rangle_{R_1} |10\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \\ &\quad \left. + 0 |10\rangle_{R_1} |11\rangle_{C_1} |01\rangle_{R_2} |11\rangle_{C_2} \right) |1\rangle_{B_1} |00\rangle_{B_2} \\ &\quad + \left(\frac{3}{16} |11\rangle_{R_1} |00\rangle_{C_1} |01\rangle_{R_2} |01\rangle_{C_2} \right. \\ &\quad + \frac{3}{16} |11\rangle_{R_1} |01\rangle_{C_1} |01\rangle_{R_2} |01\rangle_{C_2} \\ &\quad + \frac{2}{16} |11\rangle_{R_1} |10\rangle_{C_1} |01\rangle_{R_2} |01\rangle_{C_2} \\ &\quad + \frac{2}{4} |11\rangle_{R_1} |11\rangle_{C_1} |01\rangle_{R_2} |01\rangle_{C_2} \left. \right) |0\rangle_{B_1} |10\rangle_{B_2} \\ &\quad + \left(0 |01\rangle_{R_1} |00\rangle_{C_1} |11\rangle_{R_2} |11\rangle_{C_2} \right. \\ &\quad + \frac{2}{4} |01\rangle_{R_1} |01\rangle_{C_1} |11\rangle_{R_2} |11\rangle_{C_2} \\ &\quad + \frac{2}{16} |01\rangle_{R_1} |10\rangle_{C_1} |11\rangle_{R_2} |11\rangle_{C_2} \\ &\quad \left. + \frac{2}{16} |01\rangle_{R_1} |11\rangle_{C_1} |11\rangle_{R_2} |11\rangle_{C_2} \right) |0\rangle_{B_1} |01\rangle_{B_2} \\ &\quad + |g_1\rangle_{R_1 C_1 R_2 C_2 B_1 B_2} \end{aligned} \quad (\text{A3})$$

Step 4: Use C-SWAP gates to swap the states of R_1 and R_2 , as well as the states of R_1 and C_2 .

To achieve the goal of swapping the 4 th and 2 th rows of the matrix, apply C-SWAP gate operations to the terms in $|\Phi_2\rangle$ labeled by $|10\rangle_{B_2}$ and $|01\rangle_{B_2}$, swapping the states of R_1 and R_2 , as well as the states of R_1 and C_2 . For this, construct the following control operators:

$$\begin{aligned} W^{(1)} &= SWAP_{R_1, C_2} \otimes |1\rangle_{B_2^1} \langle 1|_{B_2^1} \\ &\quad + I_{R_1 C_2} \otimes |0\rangle_{B_2^1} \langle 0|_{B_2^1} \\ W^{(2)} &= SWAP_{R_1, R_2} \otimes |1\rangle_{B_2^2} \langle 1|_{B_2^2} \\ &\quad + I_{R_1 R_2} \otimes |0\rangle_{B_2^2} \langle 0|_{B_2^2} \end{aligned}$$

Apply $W_{R_1 R_2 C_2 B_2}^{(3)} = W^{(2)} W^{(1)}$ to $|\Phi_2\rangle$

$$\begin{aligned} |\Phi_3\rangle &= W_{R_1 R_2 C_2 B_2}^{(3)} |\Phi_2\rangle \tag{A4} \\ &= \frac{1}{\sqrt{3}} \left[\left(|\psi_1\rangle_{R_1 C_1} + |\psi_3\rangle_{R_1 C_1} \right) \right. \\ &\quad |01\rangle_{R_2} |11\rangle_{C_2} |1\rangle_{B_1} |00\rangle_{B_2} \\ &\quad + |\psi_2\rangle_{R_1 C_1} |01\rangle_{R_2} |11\rangle_{C_2} |0\rangle_{B_1} |10\rangle_{B_2} \\ &\quad \left. + |\psi_4\rangle_{R_1 C_1} |01\rangle_{R_2} |11\rangle_{C_2} |0\rangle_{B_1} |01\rangle_{B_2} \right] \\ &\quad + |g_2\rangle_{R_1 C_1 R_2 C_2 B_1 B_2} \end{aligned}$$

where $|\psi_i\rangle_{R_1 C_1}$, $i = 1, 2, 3, 4$ represents the first, second, third, fourth row of the target matrix, respectively:

$$\begin{aligned} |\psi_1\rangle_{R_1 C_1} &= \frac{1}{4} |00\rangle_{R_1} |00\rangle_{C_1} + \frac{1}{16} |00\rangle_{R_1} |01\rangle_{C_1} \\ &\quad + \frac{3}{16} |00\rangle_{R_1} |10\rangle_{C_1} + \frac{3}{16} |00\rangle_{R_1} |11\rangle_{C_1}, \end{aligned}$$

$$\begin{aligned} |\psi_2\rangle_{R_1 C_1} &= \frac{3}{16} |01\rangle_{R_1} |00\rangle_{C_1} + \frac{3}{16} |01\rangle_{R_1} |01\rangle_{C_1} \\ &\quad + \frac{2}{16} |01\rangle_{R_1} |10\rangle_{C_1} + \frac{2}{4} |01\rangle_{R_1} |11\rangle_{C_1}, \end{aligned}$$

$$\begin{aligned} |\psi_3\rangle_{R_1 C_1} &= \frac{7}{16} |10\rangle_{R_1} |00\rangle_{C_1} + 0 |10\rangle_{R_1} |01\rangle_{C_1} \\ &\quad + \frac{1}{4} |10\rangle_{R_1} |10\rangle_{C_1} + 0 |10\rangle_{R_1} |11\rangle_{C_1}, \end{aligned}$$

$$\begin{aligned} |\psi_4\rangle_{R_1 C_1} &= 0 |11\rangle_{R_1} |00\rangle_{C_1} + \frac{2}{4} |11\rangle_{R_1} |01\rangle_{C_1} \\ &\quad + \frac{2}{16} |11\rangle_{R_1} |10\rangle_{C_1} + \frac{2}{16} |11\rangle_{R_1} |11\rangle_{C_1} \end{aligned}$$

Step 5: Label the useful information terms and useless information terms

Introduce a 1-qubit auxiliary B_3 in the state $|0\rangle_{B_3}$, and use B_3 to label the useful and useless information terms in $|\Phi_3\rangle$. To do this, construct a control operator:

$$W_{B_1 B_2 B_3}^{(4)} = V^{(1)} V^{(2)} V^{(3)},$$

where $V^{(i)} = P_{B_1 B_2}^{(i)} \otimes \sigma_{B_3}^{(x)} + (I_{B_1 B_2} - P_{B_1 B_2}) \otimes I_{B_3}$, $P_{B_1 B_2}^{(i)}$ ($i = 1, 2, 3$) is the projection operator

$$\begin{aligned} P_{B_1 B_2}^{(1)} &= |100\rangle_{B_1 B_2 B_1 B_2} \langle 100| \\ P_{B_1 B_2}^{(2)} &= |001\rangle_{B_1 B_2 B_1 B_2} \langle 001| \\ P_{B_1 B_2}^{(3)} &= |010\rangle_{B_1 B_2 B_1 B_2} \langle 010|. \end{aligned}$$

Apply $W_{B_1 B_2 B_3}^{(4)}$ to $|\Phi_3\rangle|0\rangle_{B_3}$, obtained:

$$\begin{aligned} |\Phi_4\rangle &= W_{B_1 B_2 B_3}^{(4)} |\Phi_3\rangle|0\rangle_{B_3} \\ &= \frac{1}{\sqrt{3}} \left[(|\psi_1\rangle_{R_1 C_1} + |\psi_3\rangle_{R_1 C_1}) \right. \\ &\quad |01\rangle_{R_2} |11\rangle_{C_2} |1\rangle_{B_1} |00\rangle_{B_2} \\ &\quad + |\psi_2\rangle_{R_1 C_1} |01\rangle_{R_2} |11\rangle_{C_2} |0\rangle_{B_1} |10\rangle_{B_2} \\ &\quad \left. + |\psi_4\rangle_{R_1 C_1} |01\rangle_{R_2} |11\rangle_{C_2} |0\rangle_{B_1} |01\rangle_{B_2} \right] |1\rangle_{B_3} \\ &\quad + |\mathcal{G}_2\rangle_{R_1 C_1 R_2 C_2 B_1 B_2} \end{aligned} \tag{A5}$$

Mark the useful information terms with $|1\rangle_{B_3}$ and the useless information terms with $|0\rangle_{B_3}$.

Step 6: Use Hadamard gates for information superposition.

Apply the Hadamard operator $W_{B_1 B_2}^{(5)} = H^{\otimes 3}$ acts on B_1 and B_2 , obtained:

$$\begin{aligned} |\Phi_5\rangle &= W_{B_1 B_2}^{(5)} |\Phi_4\rangle \\ &= \frac{1}{(\sqrt{2})^3 \cdot \sqrt{3}} \left[|\psi_1\rangle_{R_1 C_1} + |\psi_3\rangle_{R_1 C_1} + |\psi_2\rangle_{R_1 C_1} \right. \\ &\quad \left. + |\psi_4\rangle_{R_1 C_1} \right] |01\rangle_{R_2} |11\rangle_{C_2} |0\rangle_{B_1} |00\rangle_{B_2} |1\rangle_{B_3} \\ &\quad + |\mathcal{G}_3\rangle_{R_1 C_1 R_2 C_2 B_1 B_2 B_3}. \end{aligned} \tag{A6}$$

Step 7: Measure the auxiliary B_1, B_2, B_3 in the state $|0001\rangle_{B_1 B_2 B_3}$.

Finally, use $|0001\rangle_{B_1 B_2 B_3}$ to measure ancilla B_1, B_2, B_3 , thus removing useless information items and obtaining useful information items as

$$|\Phi_6\rangle = |\Psi_{out}\rangle |01\rangle_{R_2} |11\rangle_{C_2},$$

where

$$|\Psi_{out}\rangle = |\psi_1\rangle_{R_1 C_1} + |\psi_2\rangle_{R_1 C_1} + |\psi_3\rangle_{R_1 C_1} + |\psi_4\rangle_{R_1 C_1}$$

The probability of the above measurement is $\frac{1}{24}$.

References

1. Benioff P (1980). The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *J. Stat. Phys.*, 22(5): 563-591.
2. Manin Y I (1980). Computable and noncomputable (in Russian). *Sov. Radio*, 13–15.
3. Deutsch D (1985). Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc. R. Soc. Lond. A*, 400(1818): 971-117.
4. Shor P W (1994). *Algorithms for quantum computation: discrete logarithms and factoring*. Proceedings 35th Annual Symposium on Foundations of Computer Science. IEEE: 124-134.
5. Shor P W (1999). Polynomial time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.*, 41(2): 303-332.

6. Grover L K (1996). *A fast quantum mechanical algorithm for database search*. Proceedings of the 28th Annual ACM Symposium on Theory of Computing. 212219.
7. Harrow A W, Hassidim A and Lloyd S (2009). Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103(15): 150502.
8. Clader B D, Jacobs B C and Sprouse C R (2013). Preconditioned quantum linear system algorithm. *Phys. Rev. Lett.*, 110(25): 250504.
9. Coppersmith D and Winograd S (1990). Matrix multiplication via arithmetic progressions. *J. Symb. Comput.*, 9(3): 251–280.
10. Weinstein Y S, Pravia M A, Fortunato E M, et al. (2001). Implementation of the quantum Fourier transform. *Phys. Rev. Lett.*, 86(9): 1889.
11. Nielsen M A, Chuang I (2000). *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press.
12. Wang H, Wu L, Liu Y, et al. (2010). Measurement based quantum phase estimation algorithm for finding eigenvalues of nonunitary matrices. *Phys. Rev. A*, 82(6):062303.
13. Motta M, Ye E, McClean J R, et al. (2021). Low rank representations for quantum simulation of electronic structure. *Npj Quantum Inf.*, 7: 83.
14. Kockum A F, Soro A and García-Álvarez L, et al. *Lecture Notes on Quantum Computing*. arXiv: 2311.08445 [quant-ph].
15. Prakash A (2014). *Quantum algorithms for linear algebra and machine learning*. Ph.D. Thesis, UC Berkeley.
16. Trotter H F (1959). On the product of semi-groups of operators. *P. Am. Math. Soc.*, 10(4): 545-551.
17. Zhao Z, Fitzsimons J K and Fitzsimons J F (2019). Quantumassisted Gaussian process regression. *Phys. Rev. A*, 99(5): 052331.
18. Stolze J, Zenchuk A I (2019). Computing scalar products via a twoterminal quantum transmission line. *Phys. Lett. A*, 383(34): 125978.
19. Wentao Q, Alexander I Z, Asutosh K and Junde W (2024). Quantum algorithms for matrix operations and linear systems of equations. *Commun. Theor. Phys.*, 76: 035103.
20. Alexander I Z, Wentao Q, Asutosh K and Junde W (2004). Matrix manipulations via unitary transformations and ancilla state measurements. *Quantum Inf. Comput.*, 24(13) : 1099-1109.
21. Alexander I Z, Georgii A B, Wentao Q, Asutosh K and Junde W (2025). Quantum Algorithms for Calculating Determinant and Inverse of Matrix and Solving Linear Algebraic Systems. *Quantum Inf. Comput.*, 25(2): 195-215.
22. Alexander I Z, Wentao Q and Junde W (2025). Matrix encoding method in variational quantum singular value decomposition. *Quantum Inf. Comput.*, 25(4): 356-368.
23. Edward B F, Alexander I Z, Wentao Q and Junde W, *Controlled measurement, Hermitian conjugation and normalization in matrix-manipulation algorithms*. arXiv:2504.00015 [quant-ph].
24. Kitaev A Y, Shen A H and Vyalı M N (2002). *Classical and Quantum Computation*, Graduate Studies in Mathematics, V.47, American Mathematical Society, Providence, Rhode Island.