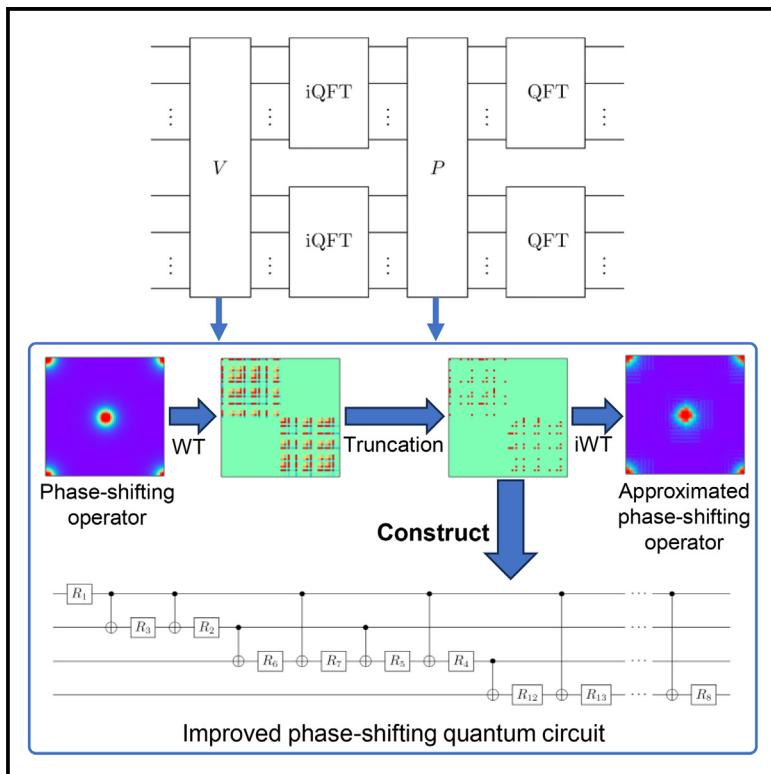


# An improved quantum algorithm of the multislice method

## Graphical abstract



## Authors

Yucheng Wang, Yang Sun, Zejun Ding

## Correspondence

yangsun@xmu.edu.cn (Y.S.),  
zjding@ustc.edu.cn (Z.D.)

## In brief

Natural sciences; Physics; Quantum theory

## Highlights

- An improved quantum algorithm of multislice method with significant speedup
- Reconstructing phase-shifting quantum circuits to eliminate multi-controlled gates
- A truncation method to further reduce gate count with controllable error
- Classical simulations to validate the feasibility and accuracy of the algorithm



## Article

# An improved quantum algorithm of the multislice method

Yucheng Wang,<sup>1</sup> Yang Sun,<sup>3,\*</sup> and Zejun Ding<sup>1,2,4,\*</sup>

<sup>1</sup>Department of Physics, University of Science and Technology of China, Hefei, Anhui 230026, P.R. China

<sup>2</sup>Hefei National Research Center for Physical Sciences at the Microscale, University of Science and Technology of China, Hefei, Anhui 230026, P.R. China

<sup>3</sup>Department of Physics, Xiamen University, Xiamen, Fujian 361005, P.R. China

<sup>4</sup>Lead contact

\*Correspondence: [yangsun@xmu.edu.cn](mailto:yangsun@xmu.edu.cn) (Y.S.), [zjding@ustc.edu.cn](mailto:zjding@ustc.edu.cn) (Z.D.)

<https://doi.org/10.1016/j.isci.2025.112165>

## SUMMARY

The multislice method is an important algorithm for electron diffraction and image simulations in transmission electron microscopy (TEM). We have proposed a quantum algorithm of the multislice method based on quantum circuit model previously. In this work, we have developed an improved quantum algorithm. We reconstruct the phase-shifting quantum circuit without using the multi-controlled quantum gates, thereby significantly improve the computation efficiency. The improved quantum circuit also allows further gate count reduction at the cost of a controllable error. We have simulated the quantum circuit on a classical supercomputer and analyzed the result to prove the feasibility and correctness of the improved quantum algorithm. We also provide proper parameter settings through testing, allowing the minimization of the necessary number of quantum gates while limiting the relative error within 1%. This work demonstrates the potential of applying quantum computing to electron diffraction simulations and achieving quantum advantages.

## INTRODUCTION

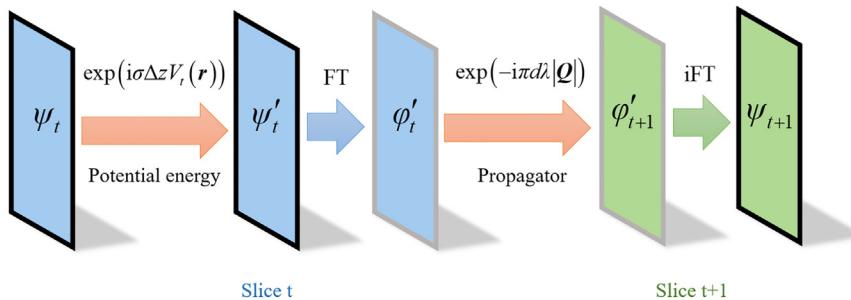
With various advancements in the physical realization of quantum computers,<sup>1–3</sup> quantum computing has received tremendous attention in recent years. By utilizing quantum entanglement and quantum superposition, quantum computing is able to efficiently solve certain complex problems that are intractable for classical computers. For example, Shor's algorithm<sup>4</sup> can solve the integer factorization problem in a polynomial time. And Grover's algorithm<sup>5</sup> can substantially speed up the search in unstructured databases. Quantum simulation<sup>6</sup> is a major application of quantum computing, with the idea, proposed by Feynman, of using a quantum system as a quantum computer to simulate another quantum system efficiently.<sup>7,8</sup>

In our previous work, we have proposed a quantum algorithm of the multislice method based on quantum circuit model to simulate electron scattering and diffraction using quantum computing.<sup>9</sup> The multislice method is a widely used method in electron microscopy to simulate the propagation of the high-energy electron beam in the three-dimensional atomic potential field. The original multislice method was proposed by Cowley<sup>10–12</sup> and has been improved and extended many times since.<sup>13–18</sup> Nowadays the multislice method has applications in various areas of electron beam related techniques, including simulations of transmission electron microscopic (TEM) images,<sup>19,20</sup> scanning transmission electron microscopic (STEM) images,<sup>21–23</sup> Bohmian quantum trajectories,<sup>24,25</sup> and electron energy loss spectra (EELS).<sup>26,27</sup> Unlike Bloch wave method,<sup>28,29</sup> the other commonly used method for simulating elec-

tron diffraction, the multislice method can in principle handle complex nonperiodic potential fields. But as object size in simulation increases, the computation cost of the classical multislice method increases exponentially due to the repeatedly use of the fast Fourier transform (FFT).<sup>30</sup> In the previous quantum algorithm, we replaced FFT with the quantum Fourier transform (QFT)<sup>31,32</sup> and constructed phase-shifting quantum circuits that work together to perform iterations, achieving a preliminary quantum speedup. One potential problem was that we used a large number of multi-controlled quantum gates in phase-shifting circuits. Current quantum processors usually have fixed topology and will have to compile multi-controlled gates into one- and two-qubit gates, which may require linear cost related to the number of control qubits.<sup>33</sup> This additional factor of cost may nullify the quantum advantage of the quantum algorithm on practical quantum hardware.

In this paper, we reconstruct phase-shifting circuits with only one- and two-qubit quantum gates, which are comparable in number to the multi-controlled gates used before. Our improved phase-shifting quantum circuit is based on the quantum circuit for diagonal unitary given by Welch et al.,<sup>34</sup> who gave a one-dimensional example of Eckart barrier in their work. We extend their approach to two-dimensional situations and apply it to the multislice method to deal with three-dimensional potential field. Then we introduce a truncation approach that allows cutting off some unimportant terms after Walsh transform to further reduce the quantum gate number. The truncation thresholds are carefully set according to an empirical formula derived from tests in order to achieve a trade-off between the quantum gate





**Figure 1.** The flowchart of the wave function iteration in the multislice method

The black frame represents the coordinate representation, while the gray one represents the momentum representation.

number and the truncation error. In this way, the improved quantum algorithm of the multislice method can use a significantly smaller number of one- and two-qubit quantum gates to perform the same simulations as before with a sufficiently small additional error.

## Theory

### Quantum multislice algorithm

The basic concept of the multislice method is to divide the sample into a series of equally spaced slices perpendicular to the electron beam. When slices are thin enough, the electron wave function can be considered to propagate freely within a slice and only interacts with a two-dimensional potential field projected on each interface. More specifically, considering one iteration of the electron wave function  $\psi(\mathbf{r})$  from the slice  $t$  to the slice  $t + 1$ , we have

$$\psi_{t+1}(\mathbf{r}) = \exp\left(-i\pi d\lambda|\mathbf{Q}|^2\right) \exp(i\sigma\Delta z V_t(\mathbf{r})) \psi_t(\mathbf{r}) \quad (\text{Equation 1})$$

where  $\sigma = \frac{me^2}{2\pi\hbar^2}$ .  $\lambda$  is the electron wavelength,  $m$  the mass of an electron,  $e$  the elementary charge,  $d$  the thickness of a slice.  $\psi_t(\mathbf{r})$  and  $V_t(\mathbf{r})$  are the wave function and the potential energy at slice  $t$ ,  $\mathbf{r}$  represents the space coordinates  $(x, y)$  in the  $xy$ -plane, and  $\mathbf{Q}$  represents the space frequency  $(Q_x, Q_y)$  in the plane. Ishizuka et al. proposed that the maximum slice thickness  $d$  should be  $\sim kd_P$  in order to get a stable result close to the limiting value, where  $k$  is the wavenumber of the incident electrons and  $d_P$  is the distance over which the potential does not change by an appreciable fraction.<sup>14</sup>

In high-energy approximation, the effect of the potential to the electron momentum can be considered as a small perturbation, hence, the kinetic energy of the electron,  $E_k$ , is always equal to the initial kinetic energy,  $eU$ , where  $U$  is the accelerating voltage of the electron beam. Considering the relativistic effect, the relativistic mass of electron can be given by

$$m = m_0 + eU/c^2 \quad (\text{Equation 2})$$

and the electron wavelength can be given by

$$\lambda = \frac{hc}{\sqrt{2E_0E_k+E_k^2}} \quad (\text{Equation 3})$$

where  $E_0 = m_0c$ ,  $m_0$  is the rest mass of electron,  $c$  is the speed of light in vacuum.

The potential energy term  $\exp(i\sigma\Delta z V_t(\mathbf{r}))$  is diagonal in the coordinate representation, and the kinetic energy term, i.e., the propagator,  $\exp(-i\pi d\lambda|\mathbf{Q}|^2)$  is diagonal in the momentum representation.

We use Fourier transform (FT), denoted by  $F$ , and inverse Fourier transform (iFT), denoted by  $F^{-1}$ , to transform the wave function between two representations. In this way, one iteration can be written as a unitary operator  $F^{-1}PFV$ , where both potential operator  $V$  and kinetic operator  $P$  are diagonal. Figure 1 show the process of one iteration.

In the quantum algorithm of the multislice method, qubits and quantum gates form a quantum circuit. Input information is encoded into the quantum state of qubits and processed by quantum gates to achieve calculation. We use two-dimensional amplitude encoding to store a slice of the wave function. The two-dimensional wave function is discretized into a  $N \times N$  complex matrix where  $N = 2^n$ . Each matrix element represents the amplitude of the wave function at a coordinate in the real space. To store this matrix, we use a quantum register with  $2n$  qubits, which has  $2^{2n}$  basis states and stores each matrix element by the amplitude of a basis state. More specifically, we divide the quantum register into two parts, each with  $n$  qubits. According to the binary code, the lower  $n$  qubits represent the  $x$ -coordinate and the higher  $n$  qubits represent the  $y$ -coordinate. In this way, each basis state corresponds to a coordinate point. Then we can use the QFT to achieve the Fourier transform in the iteration, which is exponentially faster than the FFT that is typically used in the classical multislice method. The quantum circuit for one-dimensional QFT is as Figure 2. Here,  $H$  represents the Hadamard gate, and  $R_m$  represents a controlled phase-shifting gate corresponding to the transformation matrix.

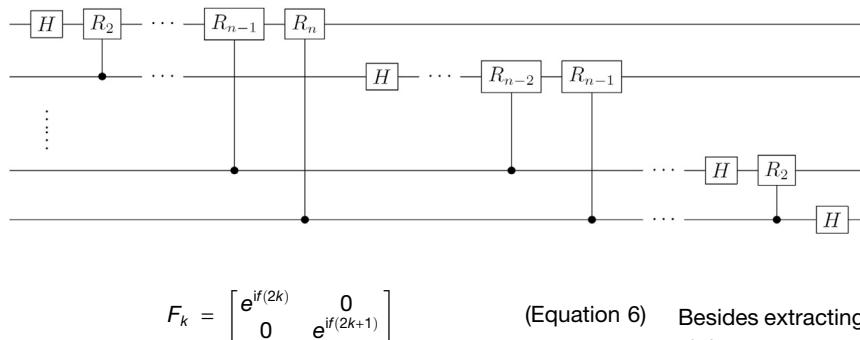
$$R_m = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i / 2^m} \end{bmatrix} \quad (\text{Equation 4})$$

The circuit can be run in reverse to perform the inverse quantum Fourier transform (iQFT). It can be proven that we can perform a two-dimensional FT on the quantum state matrix by applying two  $n$ -qubit iQFT to both lower  $n$  qubits and higher  $n$  qubits in parallel.<sup>9</sup>

Apart from QFT circuits and iQFT circuits, there are phase-shifting quantum circuits that apply diagonal phase-shifting operator on the quantum state, satisfying:

$$U|x\rangle = e^{i\phi(x)}|x\rangle \quad (\text{Equation 5})$$

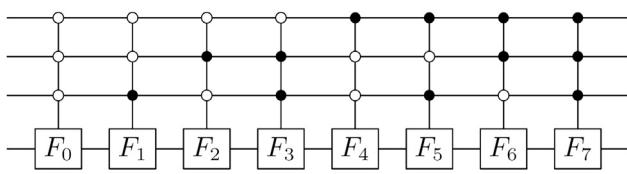
In the previous work, we used a simple phase-shifting circuit shown in Figure 3 to implement this operator, where  $k \in \{0, 1, \dots, 2^{2n-1} - 1\}$ , and  $F_k$  corresponds to the matrix



Then the whole quantum circuit for one iteration is as shown **Figure 4**, where  $V$  and  $P$  represent respectively the phase-shifting quantum circuits for the potential operator and the free propagation operator. By preparing an initial state on qubits and applying this circuit iteratively, we can simulate the evolution of the electron wave function. We use the most basic input state, plane wave, as an example. To prepare this initial state, we only need an all-zero state and then apply a Hadamard gate on each qubit to get a uniform superposition state. Other arbitrary forms of initial quantum states can be obtained by a sublinear quantum circuit with complexity no more than  $O(N)$ ,<sup>35</sup> and there is a more efficient method for a sparse state.<sup>36</sup> After a certain number of iterations, the position distribution and momentum distribution of the electron can be extracted from the output quantum state.

Notice that the phase-shifting quantum circuit utilizes a large number of multi-controlled quantum gates as we can see in **Figure 3**, which may be inefficient for practical near-term quantum hardware.<sup>33</sup> In the next section, we will focus on the optimization of the phase-shifting circuit for addressing this problem.

In practical quantum computing, how to efficiently extract information from the quantum state is also a problem worthy of study. A basic approach is to perform a standard projective measurement after each run and to repeat the quantum simulation. Each run provides a single sample, and the measured results will form a distribution which matches with the probability distribution of the electron state. Here, the number of repetitions needed depends on the shape of the distribution and the statistical error we can accept. A sampling demonstration is presented in our previous work,<sup>9</sup> where we reconstruct the electron diffraction pattern by using up to 5,000 samples in the momentum representation. In this case, the required sample number is much smaller than the number of grid points and does not increase with  $N$ , adding only a constant factor to the complexity, because in the momentum representation the



**Figure 3.** The previous version of the phase-shifting quantum circuit for 4 qubits ( $n = 2$ )

**Figure 2.** The quantum circuit for 1D QFT

samples have a high probability of landing on the fixed diffraction peaks and forming the pattern easily. Advanced extraction of full quantum state would involve quantum state tomography<sup>37</sup> which will not be discussed in this paper.

Besides extracting information directly from the output quantum state, one may use the output quantum state as input for further possible quantum algorithms to bypass the full extraction. One may also use it for a fast sampling of electron probability distribution as a sub-algorithm of other simulations.

#### Optimization of the phase-shifting circuit

We reconstruct the phase-shifting quantum circuit by using the Walsh transform.<sup>38</sup> The Walsh transform expands a discrete function by a series of orthogonal basis functions named Walsh functions. For a 1D function  $f(x)$  where  $x \in \{0, 1, \dots, N - 1\}$ ,  $N = 2^n$ , the Walsh functions in Hadamard order are defined as follow:

$$T_u(x) = (-1)^{\sum_{i=1}^{n-1} x_i u_i} \quad (\text{Equation 7})$$

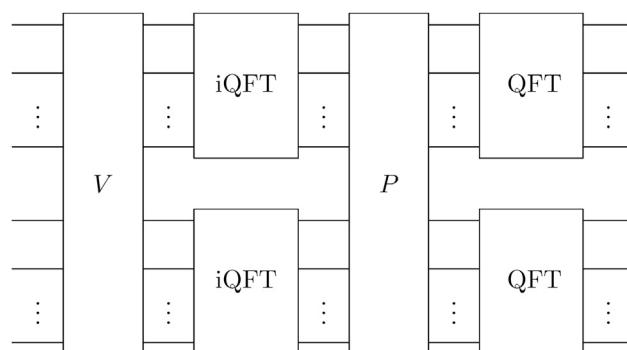
where  $u \in \{0, 1, \dots, N - 1\}$ , and  $x_n$  represents the  $n$ th bit in the binary expansion of  $x$ , i.e.,  $x = (x_{n-1}x_{n-2}\dots x_0)_2$ ,  $u = (u_{n-1}u_{n-2}\dots u_0)_2$ . Under this definition, the transform is named Walsh-Hadamard transform, which can be written as follow:

$$W(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) (-1)^{\sum_{i=0}^{n-1} x_i u_i} \quad (\text{Equation 8})$$

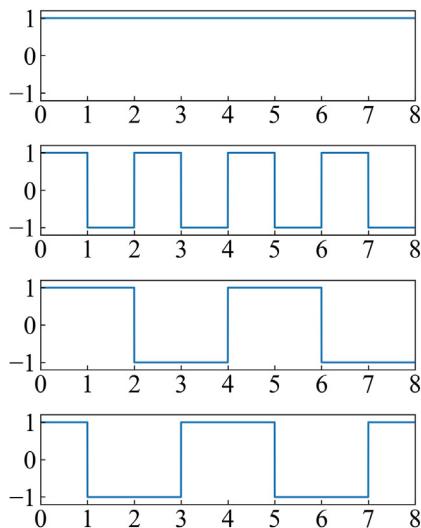
And  $f(x)$  can be represented as a linear combination of Walsh basis functions, called Walsh expansion:

$$f(x) = \sum_{u=0}^{N-1} W(u) T_u(x) \quad (\text{Equation 9})$$

For example, the 1D Walsh basis functions when  $N = 8$  are shown in **Figure 5**.

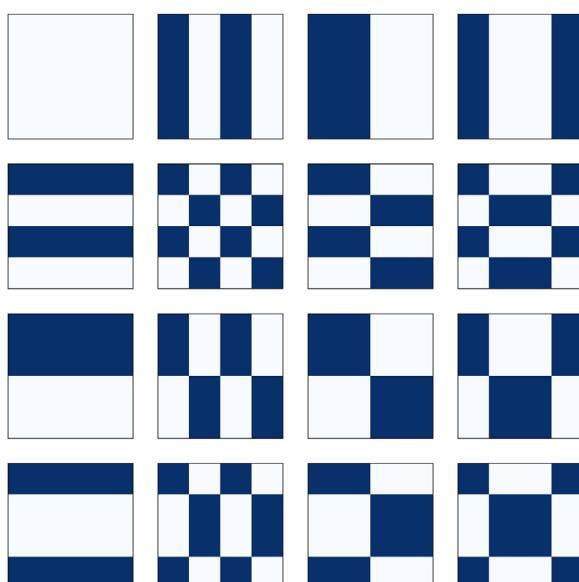


**Figure 4.** Diagram of the quantum circuit for the iteration of one slice



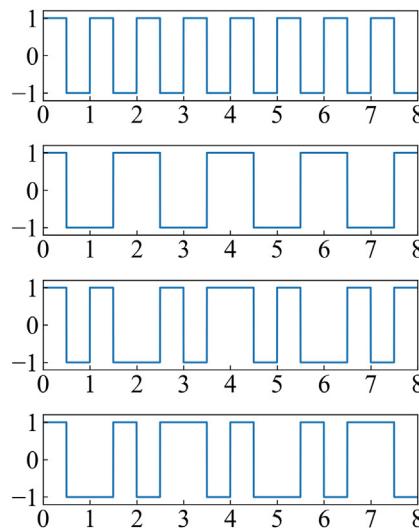
For 2D situation, we just need to combine the binary expansions of  $x$  and  $y$  together, and apply the 1D formula. For example, the 2D Walsh basis function when  $N = 4$  are shown in Figure 6. Similar to the 1D situation, an arbitrary  $4 \times 4$  discrete function can be seen as a linear superposition of these 16 basis functions.

Interestingly, the patterns of Walsh functions are well suited for a quantum circuit. In our previous version of the phase-shifting quantum circuit, we were actually manipulating the phase of each basis state individually, which is hard because of the superposition. Multiple control qubits must be attached to the phase-shifting quantum gate as shown in Figure 3 to shift the phase of a single basis state, while a phase-shifting gate without control qubits will result in a phase shift for half of the basis states, forming a pattern that interestingly just matches a



**Figure 6. The 2D  $4 \times 4$ -point Walsh functions**

The white pixel represents 1 while the blue pixel represents -1.



**Figure 5. The 1D 8-point Walsh functions in Hadamard order**

Walsh function. We can then construct a phase-shifting quantum circuit for the phase-shifting operator  $|x\rangle \rightarrow e^{if(x)}|x\rangle$  according to the Walsh transform of  $f(x)$ . An 1D example has been given by Welch et al.,<sup>34</sup> and we extend their approach to two-dimensional situations.

The improved phase-shifting quantum circuit for 4 qubits is shown in Figure 7 as an example, consisting of  $N^2 - 1$  phase-shifting quantum gates and a number of controlled-NOT (CNOT) quantum gates between them, with no multi-controlled gate used.

Each phase-shifting gate corresponds

to one term of the Walsh expansion of the phase-shifting operator, i.e., a scaled Walsh function with a certain index. The amount of the phase shift is determined by each coefficient of this Walsh function. The first Walsh function with index 0 corresponds to a global phase shift, which is generally considered to be meaningless in quantum physics. These phase-shifting gate are ordered according to the Gray code<sup>39</sup> to minimize the number of CNOT gates required. Gray code is a binary code system where two successive values differ in only one bit. CNOT gates between two phase-shifting gates are determined by the difference in the binary expansions of the two neighboring indexes, i.e., the Hamming distance.<sup>40</sup> In this order, only one CNOT gate is required between every two phase-shifting gates.

This phase-shifting quantum circuit can be generalized to two dimensions. In our algorithm, the wave function of a slice is stored as a  $N \times N$  matrix with two indexes  $x$  and  $y$ . We can concatenate the binary codes of two indexes together and decode it as a new index  $r$ ,  $r \in \{0, 1, \dots, N^2 - 1\}$ , which satisfies:

$$r = Ny + x \quad (\text{Equation 10})$$

In this way, we reshape a 2D matrix into a 1D array while the correspondence between the phase shifts that need to be applied to each element of the wave function remains. Specifically, to apply a 2D phase-shifting operator satisfying:

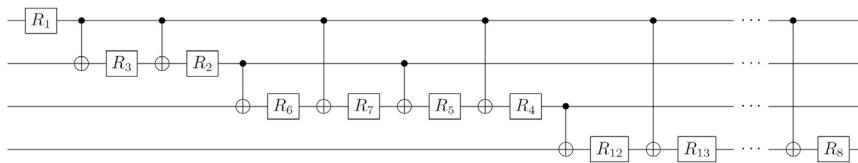
$$U|x,y\rangle = e^{if(x,y)}|x,y\rangle \quad (\text{Equation 11})$$

we set a  $g(r)$  satisfying:

$$g(r) = f(x,y) \quad (\text{Equation 12})$$

where  $x = r \bmod N$ ,  $y = r - Nx$ . Then applying  $U$  to the 2D matrix of the wave function is equivalent to applying  $U'$  to the reshaped 1D array, where  $U'$  satisfies:

$$U'|r\rangle = e^{ig(r)}|r\rangle \quad (\text{Equation 13})$$



We can represent  $U'$  as a  $N^2 \times N^2$  diagonal matrix:

$$U' = \text{diag}(e^{if(0,0)}, e^{if(1,0)}, \dots, e^{if(N,0)}, e^{if(0,1)}, \dots, e^{if(N,1)}, \dots, e^{if(N,N)}) \quad (\text{Equation 14})$$

The quantum state in qubits need no change for the reshaping, it can be naturally decoded in two ways because of our encoding method where the lower  $n$  qubits represent the  $x$ -coordinate and the higher  $n$  qubits represent the  $y$ -coordinate.

We also need to apply Walsh transform to the potential operator matrices and the kinetic operator matrix as a classical preprocessing step to determine the parameters in the phase-shifting quantum circuits. The complexity of the fast Walsh-Hadamard transform (FWHT) is  $O(N \log N)$ , which is the same as the complexity of the FFT. Nevertheless, this preprocessing step is efficient and will not become a new bottleneck of the whole algorithm. First, unlike the FFT, which uses a large number of multiplications, the FWHT uses only additions, which are much faster for classical computers. What is more, the results obtained from the Walsh transform are reusable. For the potential field in the lattices, we only need to compute the Walsh transform of the potential field slices within one lattice. The computational cost does not increase continually with the number of lattice layers. It can also be reused for different input electron states. The kinetic term is even simpler and remains constant, requiring only one Walsh transform to be able to reuse throughout the entire simulation process.

In summary, our improved quantum algorithm still follows the diagram in Figure 4, using the improved phase-shifting quantum sub-circuit for  $2n$  qubits to achieve the phase-shifting operations  $V$  and  $P$ . The parameters of the quantum circuit are determined by the WHT preprocessing steps for operators  $V$  and  $P$ , which run on a classical computer.

So far, the quantum algorithm has been an exact quantum version of the classical multislice method. However, if we allow for some additional error, we can further reduce the number of quantum gates required. After the Walsh transform of the operator, we set a truncation threshold and cut off the terms with coefficients that are less than the threshold. In this way, we can ignore and skip those Walsh basis functions with low contributions and achieve an approximation of the operator using a quantum circuit with lower depth. It is worth mentioning that this optimization can only be done in the quantum algorithm, not in the classical algorithm. In the classical algorithm, without the quantum superposition, operating half of the phases to execute a phase shift that matches a Walsh function pattern is inefficient. It is also not easy to separate the components of each Walsh basis function in the operator and skip some of them.

In this paper, we use a relative truncation threshold  $\tau$  and cut off a term in Walsh decomposition if its coefficient  $w$  is lower than

**Figure 7.** The improved version of the phase-shifting quantum circuit for 4 qubits ( $n = 2$ )

$\tau w_{\max}$  where  $w_{\max}$  represents the maximum coefficient in the Walsh decomposition and  $\tau \in [0, 1]$ . As  $\tau$  increases, the number of required quantum gates will decrease, while the truncation error will increase. We test the quantum algorithm with different qubit number and truncation threshold settings and obtain an empirical formula to appropriately set the truncation threshold separately for the potential energy terms and the kinetic energy terms, ensuring that the truncation error is kept within a certain range while minimizing the number of required quantum gates. Detailed truncation threshold settings and error analysis will be discussed together with the feasibility verification test in the next section.

## RESULTS AND DISCUSSION

### Correctness verification for the improved quantum algorithm

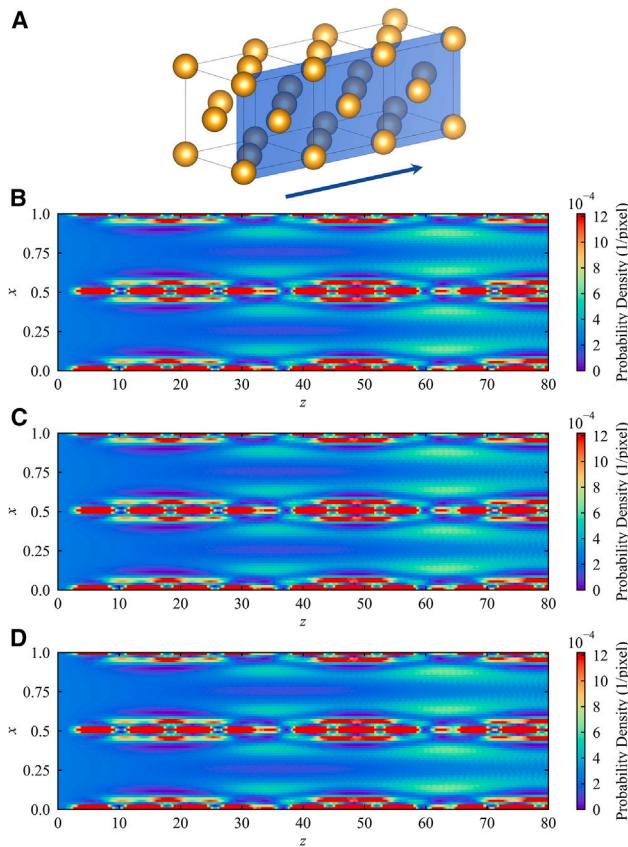
Current quantum hardware does not yet have enough qubits and sufficient fidelity to run this quantum algorithm, so we use a classical supercomputer to simulate a quantum circuit with a Python package “pyqpanda”<sup>41,42</sup> at the level of qubit and quantum gate, and run the quantum algorithm on the virtual quantum circuit. We simulate the same scenario as in the previous work, where electrons at different energies incident into a thick Au specimen, to facilitate a comparison between the previous quantum algorithm and the improved one.

The following formula is used to calculate the atomic potential field in the crystal<sup>43</sup>:

$$\varphi(\mathbf{r}) = \frac{\hbar^2}{2\pi m_0 e} \sum_i a_i \left( \frac{4\pi}{b_i + B} \right)^{3/2} \exp\left(-\frac{4\pi^2 r^2}{b_i + B}\right) \quad (\text{Equation 15})$$

where  $a_i$  and  $b_i$  ( $i = 1, 2, 3, 4$ ) are fitting parameters,  $B$  is the Debye-Waller factor that depends on temperature.

To verify the correctness of the improved quantum algorithm of the multislice method, we have simulated electron diffraction using the same parameters with three different algorithms: the classical multislice algorithm, the previous quantum algorithm, and the present improved one, obtaining identical results. Here, we keep all the terms in the improved phase-shifting part to avoid any truncation error. The calculation results can be extracted from different perspectives, and here, we choose to display the side cross section of the electron probability density distribution for easy comparison. The position of the cross section is shown in Figure 8A, and the simulation results are shown in Figures 8B–8D. This indicates that the improved phase-shifting quantum circuit can accurately replace the previous version, thereby avoiding the use of multi-controlled quantum gates.



**Figure 8. The comparison of simulation results of the three algorithms**

(A) The diagram of a side cross section atomic plane parallel to the electron beam direction. The arrow indicates the incident direction of the electron beam.  
 (B-D) The simulation results of three algorithms for plane wave incident electrons at 100 keV: (B) the previous quantum algorithm, (C) the improved quantum algorithm, (D) the classical algorithm. We use lattice constant as the unit length. Each unit cell is divided by 16 slices in z-direction, satisfying the limitation of the maximum slice thickness.  $n = 6$ , meaning 12 qubits are used in the quantum algorithm.

#### Walsh transform and truncation optimization

Next, we test different truncation thresholds based on this set of parameters. As described in the previous section, we apply the Walsh-Hadamard transform to the potential operators and the kinetic operator, then we cut off the terms of the Walsh decompositions with coefficients that are less than  $\tau w_{\max}$ .

Figure 9 visually shows the effect of Walsh transform and truncation on the operator matrices. There are 16 different potential operators for the 16 slices per cell, and we choose the one on the atomic plane. Meanwhile, the kinetic operator is the same for every slice. It is clear from Figure 9 that as the truncation threshold increases, the approximation of the operator becomes increasingly distorted because more terms are cut off.

Figure 10 shows the effect of the truncation on the simulation results. Similarly, we use the side cross section of the electron probability density distribution to illustrate. It can be seen that as the truncation threshold increases, the simulation results also become increasingly distorted, being consistent with Figure 9.

It is worth noting that the distribution patterns of the Walsh coefficients for the potential operator and the kinetic operator differ significantly as shown in Figure 9. For the potential operator, about half of the coefficients are close to zero because of its symmetry, as shown in green. The remaining coefficients have a relatively scattered distribution in terms of their magnitudes. As for the kinetic operator, due to its simpler analytical expression, almost all coefficients are zero, with only a few non-zero coefficients distributed along the first row and first column. This indicates that much fewer Walsh basis functions are required to approximate the kinetic operator than the potential operator.

#### Setting of truncation thresholds and analysis of truncation errors

To further explore suitable truncation thresholds for different operators, we then cut off potential terms or kinetic terms separately with different truncation thresholds and analysis truncation errors quantitatively. We use the average relative error defined as follow to quantify the truncation error:

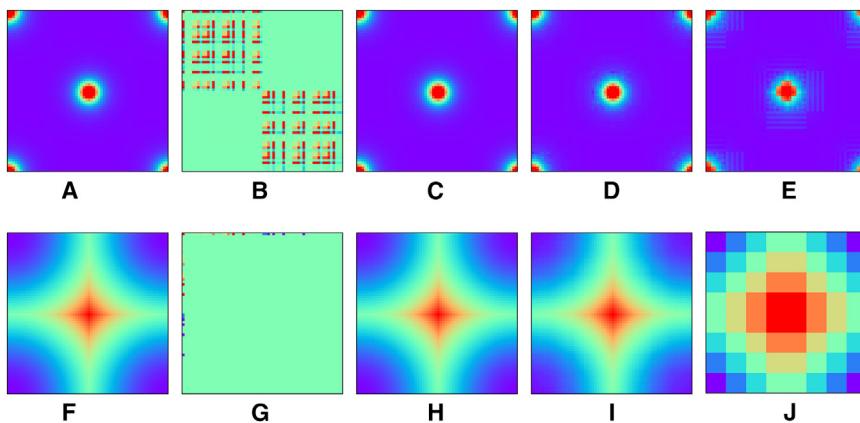
$$\epsilon = \frac{\sum |x - \hat{x}|}{\sum |\hat{x}|} \quad (\text{Equation 16})$$

where  $x$  represents the approximate value and  $\hat{x}$  represents the exact value.

We start with an initial screening where all zero terms are removed and then we apply further truncation separately to either kinetic or potential terms. During the testing, we first keep all the non-zero kinetic terms while gradually increasing the truncation threshold  $\tau_V$  for potential terms and denote the total number of the remaining terms as  $s_V$ . Similarly, we keep all the non-zero potential terms while gradually increasing the truncation threshold  $\tau_P$  for kinetic terms and denote the total number of the remaining terms as  $s_P$ . The results are shown in Figure 11. The number of quantum gates in the quantum circuit is proportional to the number of the remaining terms, so fewer terms mean higher efficiency. We can see that as  $\tau_V$  increases, the truncation error increases and the term number decreases, both changing linearly and steadily in log scale, making it very suitable for the trade-off between the two. However, as  $\tau_P$  increases, the term number only slightly decreases, while the truncation error rises sharply to an unacceptable level after a certain point. This is because that the coefficients of the potential terms are distributed more evenly and widely, while most coefficients of the kinetic terms are zero and the remaining few Walsh terms contribute so significantly that they cannot be ignored.

Therefore, a reasonable strategy is to keep all non-zero kinetic terms, which are already quite a few in number, and set an appropriate threshold for the potential terms to balance the error and the number of remaining terms. We also need to consider that the distributions of the coefficients vary with the qubit number.

Figure 12 shows the truncation error and the total number of the remaining terms with different truncation thresholds when using 12, 14, 16 qubits. Overall, to maintain consistent relative error, as the qubit number increases, the threshold needs to be appropriately lowered to keep more terms. Here, we provide an empirical formula by the three sample points closest to the

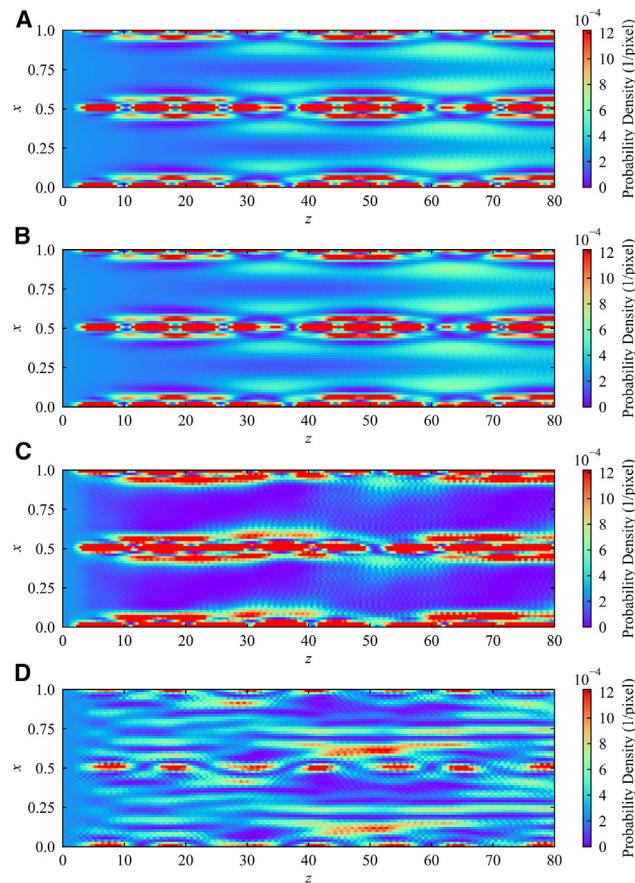


**Figure 9. The effect of Walsh transform and truncation on the operators**

- (A) The potential operator.
- (B) The potential operator after Walsh transform.
- (C–E) The approximate potential operators when  $\tau$  are (C) 0.001, (D) 0.01, (E) 0.1.
- (F) The kinetic operator.
- (G) The kinetic operator after Walsh transform.
- (H–J) The approximate kinetic operators when  $\tau$  are (H) 0.001, (I) 0.01, (J) 0.1.

dashed line in Figure 12 to maintain approximately a 1% average relative error:

$$\tau_V = \frac{128}{2^n} \times 10^{-3} \quad (\text{Equation 17})$$



**Figure 10. The effect of the truncation on the simulation results**

The figure shows the simulation results of the electron probability density distribution using the same truncation threshold  $\tau$  for the kinetic operator and the potential operator: (A)  $\tau = 0$  (exact); (B)  $\tau = 0.001$ ; (C)  $\tau = 0.01$ ; (D)  $\tau = 0.1$ .

Of course, one can tweak this truncation threshold for a faster computation speed or a lower truncation error.

We use this set of better truncation thresholds to plot the simulation results for different numbers of qubits and count the total number of quantum gates, comparing it with the situations without truncation optimization. The truncation threshold for the propagator,  $\tau_P$ , is set to  $10^{-10}$ , in order to keep all the non-zero terms. And the truncation threshold for the potential operator,  $\tau_V$ , is set according to Equation 17. The simulation results are shown in Figure 13. It can be seen that, with a relative error of about 1%, the approximate results are visually very close to the exact results, and keep consistent with different qubit number.

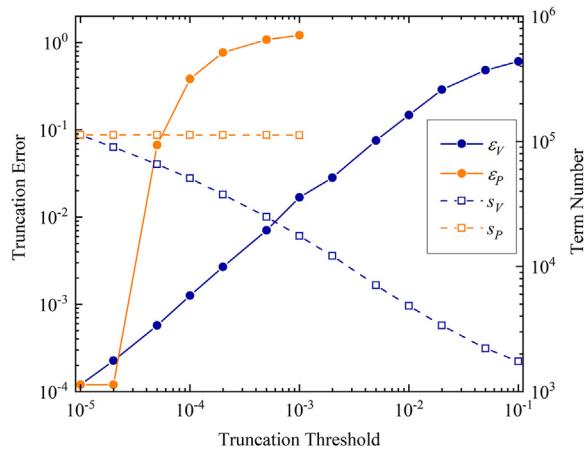
#### Computational efficiency improvement through truncation

Since as the number of qubits increases, we need to lower the truncation threshold to keep more potential terms, there is a possibility that, as the number of qubits continues to increase in larger simulations, the speedup of truncation may diminish. To examine this problem, we calculate the percentage of the remaining terms to the original terms at different qubit number, based on the empirical formula aforementioned to determine the truncation threshold. The results are shown in Figure 14. As can be seen from the figure, although the truncation threshold is lowered and we keep more terms when the qubit number increases, the percentage of the remaining terms actually decreases. This indicates that the aforementioned problem does not exist; instead, the speedup effect of the truncation will improve as we use more qubits.

We also count the number of quantum gates to quantify the computational cost of the quantum algorithm, as shown in Figure 15. We can see that through truncation optimization, we have successfully reduced the number of quantum gates by more than one order of magnitude with controllably small error. Moreover, the slope of the line after truncation optimization is lower, indicating that the speedup effect may increase as the size of the quantum circuit grows.

#### Conclusion

In this paper, we present an improved quantum algorithm of the multislice method, which can be used to simulate the 3D scattering and diffraction of high-energy electrons with quantum circuits. This improved quantum algorithm is based on our previous version, with a focus on optimizing the



**Figure 11.** The truncation errors and the number of the total remaining terms when a gradually increasing truncation threshold is applied to either potential or kinetic terms

$\epsilon_V$  and  $\epsilon_P$  represent the truncation errors for potential and kinetic terms, and  $s_V$  and  $s_P$  represent the number of the total remaining terms after the potential and kinetic truncation, respectively. Here  $n = 8$ , meaning 16 qubits are used in the quantum algorithm.

phase-shifting circuit. Using the Walsh transform, we reconstructed the phase-shifting quantum circuit by replacing all multi-controlled gates with a comparable number of one-qubit and two-qubit gates, thereby addressing the potential performance issue of the previous quantum algorithm on practical quantum hardware.

The improved phase-shifting circuit also offers the possibility of further reducing the computational cost by truncating the Walsh terms with small coefficients. Through sufficient testing using different parameters, we have found a suitable truncation threshold setting scheme that can further reduce the number of quantum gates required by more than an order of magnitude while keeping the relative error on the order of 1%. And we show that this speedup effect is not weakened

but enhanced as the number of qubits increases, ensuring that this truncation optimization can be used for larger-scale quantum simulations.

In our previous work, without considering the compilation issue of multi-controlled gates, we had initially achieved a complexity advantage relative to the classical multislice algorithm. And after the optimizations in this paper, the improved phase-shifting circuit, combined with the QFT which replaces the FFT, makes the overall complexity advantage of the improved quantum algorithm clearer. This demonstrates the potential application of quantum computing in the field of electron scattering and diffraction simulations and brings us closer to achieving quantum advantage using real quantum hardware in this area.

### Limitations of the study

While our improved quantum algorithm for the multislice method demonstrates significant advancements in computational efficiency and potential quantum advantage, it is critical to recognize several limitations of this study. Firstly, the performance of current quantum processors is insufficient to practically run the quantum algorithm in this paper. Therefore, our simulations are based on classical supercomputers. And achieving a practical quantum advantage will require further advances in quantum hardware. Secondly, how to efficiently extract and utilize the simulation results encoded in quantum states on real quantum processors still needs further research. Thirdly, the truncation thresholds for the truncation method are not universal and need to be tuned according to the specific problem and the accuracy needed.

### RESOURCE AVAILABILITY

#### Lead contact

Requests for further information and resources should be directed to and will be fulfilled by the lead contact, Prof. Zejun Ding ([zjding@ustc.edu.cn](mailto:zjding@ustc.edu.cn)).

#### Materials availability

This study did not generate new unique reagents.

#### Data and code availability

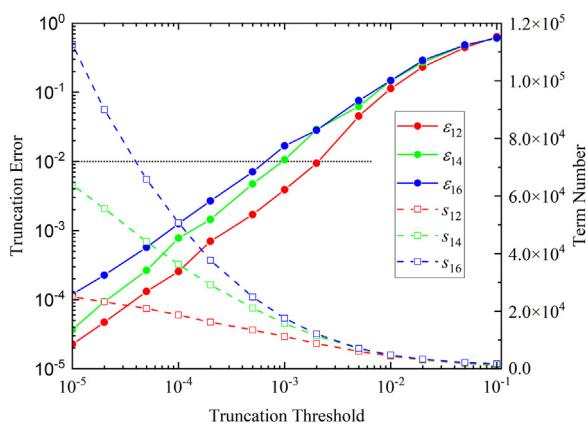
- All data reported in this paper will be shared by the [lead contact](#) upon request.
- All original code is available at [micro.ustc.edu.cn/sourcecode/QMSV2.zip](http://micro.ustc.edu.cn/sourcecode/QMSV2.zip).
- Any additional information required to reanalyze the data reported in this paper is available from the [lead contact](#) upon request.

### ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (grants nos. T2422016 and 42374108), the Fundamental Research Funds for the Central Universities (grant no. 20720230014) and the Natural Science Foundation of Xiamen (grant no. 3502Z202371007). We thank Prof. H.M. Li and the supercomputing center of USTC for the support of parallel computing.

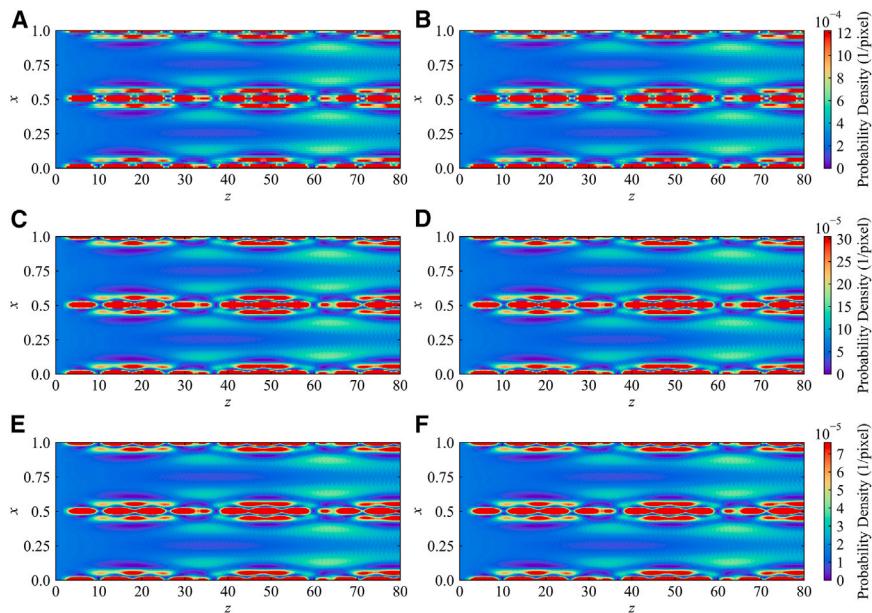
### AUTHOR CONTRIBUTIONS

Conceptualization ideas, Z.D.; methodology, Y.W.; software, Y.W. and Y.S.; formal analysis, Y.W.; investigation, Y.W.; writing – original draft, Y.W.; writing – review and editing, Z.D.; visualization, Y.W.; supervision, Z.D.; project administration, Z.D.; funding acquisition, Y.S.



**Figure 12.** The truncation error and the total term number with different truncation thresholds for 12, 14, and 16 qubit number

The truncation error is denoted by  $\epsilon$  and the total term number is denoted by  $s$ .  $\epsilon_{12}, s_{12}$ : 12 qubits,  $n = 6$ ;  $\epsilon_{14}, s_{14}$ : 14 qubits,  $n = 7$ ;  $\epsilon_{16}, s_{16}$ : 16 qubits,  $n = 8$ .



**Figure 13. The comparison of simulation results before and after truncation optimization**

(A, C, and E) The exact simulation results without truncation and (B, D, and F) the approximate simulation results using truncation optimization with different number of qubits: (A) and (B) 12 qubits, (C) and (D) 14 qubits, (E) and (F) 16 qubits.  $\tau_P$  is set to  $10^{-10}$  in (B) (D) (F),  $\tau_V$  is set to: (B) 0.002, (D) 0.001, (F) 0.0005.

#### DECLARATION OF INTERESTS

The authors declare no competing interests.

#### STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

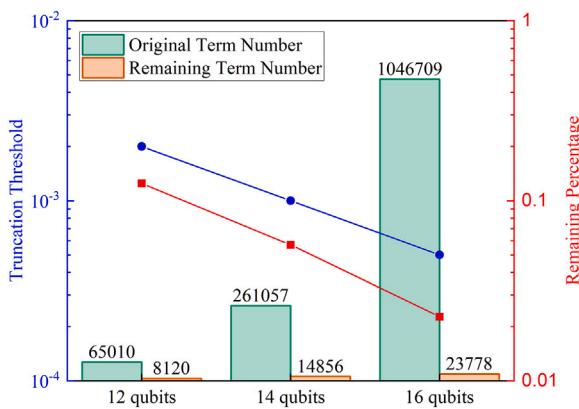
- KEY RESOURCES TABLE
- EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS
- METHOD DETAILS
- QUANTIFICATION AND STATISTICAL ANALYSIS

Received: December 2, 2024

Revised: January 6, 2025

Accepted: February 28, 2025

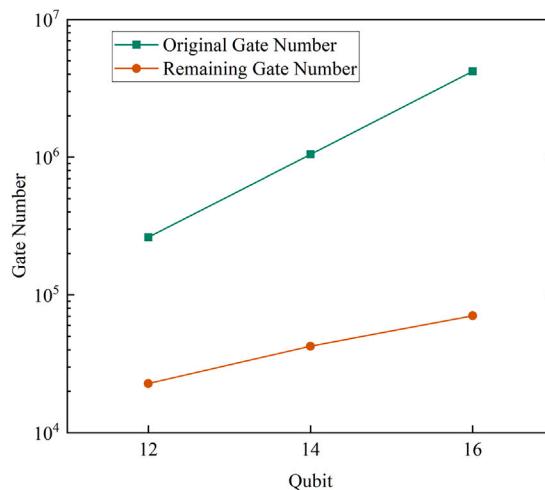
Published: March 4, 2025



**Figure 14. The remaining percentage of terms and the truncation thresholds, ensuring that the truncation error is controlled around 1%**

#### REFERENCES

- Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J.C., Barends, R., Biswas, R., Boixo, S., Branda, F.G.S.L., Buell, D.A., et al. (2019). Quantum supremacy using a programmable superconducting processor. *Nature* 574, 505–510. <https://doi.org/10.1038/s41586-019-1666-5>.
- Wu, Y., Bao, W.S., Cao, S., Chen, F., Chen, M.C., Chen, X., Chung, T.H., Deng, H., Du, Y., Fan, D., et al. (2021). Strong Quantum Computational Advantage Using a Superconducting Quantum Processor. *Phys. Rev. Lett.* 127, 180501. <https://doi.org/10.1103/PhysRevLett.127.180501>.
- Zhong, H.S., Deng, Y.H., Qin, J., Wang, H., Chen, M.C., Peng, L.C., Luo, Y.H., Wu, D., Gong, S.Q., Su, H., et al. (2021). Phase-Programmable Gaussian Boson Sampling Using Stimulated Squeezed Light. *Phys. Rev. Lett.* 127, 180502. <https://doi.org/10.1103/PhysRevLett.127.180502>.



**Figure 15. Quantum gate numbers before and after truncation optimization**

4. Shor, P.W. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In 35th Annual Symposium on Foundations of Computer Science, Santa Fe, S. Goldwasser, ed. (IEEE Computer Society). <https://doi.org/10.1109/sfcs.1994.365700>.
5. Grover, L.K. (1996). A fast quantum mechanical algorithm for database search. In Proceedings of 1996 ACM Symposium on Theory of Computing, Philadelphia (Association for Computing Machinery). <https://doi.org/10.1145/237814.237866>.
6. Georgescu, I.M., Ashhab, S., and Nori, F. (2014). Quantum simulation. *Rev. Mod. Phys.* 86, 153–185. <https://doi.org/10.1103/RevModPhys.86.153>.
7. Feynman, R.P. (1982). Simulating physics with computers. *Inter. J. Theor. Phys.* 21, 467–488. <https://doi.org/10.1007/bf02650179>.
8. Feynman, R.P. (1986). Quantum mechanical computers. *Found. Phys.* 16, 507–531. <https://doi.org/10.1007/bf01886518>.
9. Wang, Y.C., Sun, Y., and Ding, Z.J. (2024). A quantum algorithm of multi-slice method. *Results Phys.* 62, 107834. <https://doi.org/10.1016/j.rinp.2024.107834>.
10. Cowley, J.M., and Moodie, A.F. (1957). The scattering of electrons by atoms and crystals. I. A new theoretical approach. *Acta Crystal.* 10, 609–619. <https://doi.org/10.1107/s0365110x57002194>.
11. Cowley, J.M., and Moodie, A.F. (1959). The scattering of electrons by atoms and crystals. II. The effects of finite source size. *Acta Crystal.* 12, 353–359. <https://doi.org/10.1107/s0365110x59001098>.
12. Cowley, J.M., and Moodie, A.F. (1959). The scattering of electrons by atoms and crystals. III. Single-crystal diffraction patterns. *Acta Crystal.* 12, 360–367. <https://doi.org/10.1107/s0365110x59001104>.
13. Goodman, P., and Moodie, A.F. (1974). Numerical evaluation of N-beam wave functions in electron scattering by the multi-slice method. *Acta Cryst. A* 30, 289–290. <https://doi.org/10.1107/S056773947400057X>.
14. Ishizuka, K., and Uyeda, N. (1977). A new theoretical and practical approach to the multislice method. *Acta Cryst. A* 33, 740–749. <https://doi.org/10.1107/s0567739477001879>.
15. van Dyck, D. (1979). Improved methods for the high speed calculation of electron microscopic structure images. *Phys. Stat. Sol.* 52, 283–292. <https://doi.org/10.1002/pssa.2210520131>.
16. Dyck, D.V. (1980). Fast computational procedures for the simulation of structure images in complex or disordered crystals: a new approach. *J. Microsc.* 119, 141–152. <https://doi.org/10.1111/j.1365-2818.1980.tb04084.x>.
17. Ishizuka, K. (1982). Multislice formula for inclined illumination. *Acta Cryst. A* 38, 773–779. <https://doi.org/10.1107/s0567739482001594>.
18. Kilaas, R., O'Keefe, M.A., and Krishnan, K.M. (1987). On the inclusion of upper Laue layers in computational methods in high resolution transmission electron microscopy. *Ultramicrosc.* 21, 47–61. [https://doi.org/10.1016/0304-3991\(87\)90006-4](https://doi.org/10.1016/0304-3991(87)90006-4).
19. Dyck, D.V. (1985). Image Calculations in High-Resolution Electron Microscopy: Problems, Progress, and Prospects. *Adv. Imag. Elect. Phys.* 65, 295–355. [https://doi.org/10.1016/s0065-2539\(08\)60880-x](https://doi.org/10.1016/s0065-2539(08)60880-x).
20. O'Keefe, M.A., and Kilaas, R. (1994). Users guide to NCEMSS. Version 2.0. (USDOE). <https://doi.org/10.2172/10196832>.
21. Ishizuka, K. (2001). Prospects of atomic resolution imaging with an aberration-corrected STEM. *J. Electron. Microsc.* 50, 291–305. <https://doi.org/10.1093/jmicro/50.4.291>.
22. Ishizuka, K. (2001). A practical approach for STEM image simulation based on the FFT multislice method. *Ultramicrosc.* 90, 71–83. [https://doi.org/10.1016/s0304-3991\(01\)00145-0](https://doi.org/10.1016/s0304-3991(01)00145-0).
23. Kim, S., Oshima, Y., Sawada, H., Kaneyama, T., Kondo, Y., Takeguchi, M., Nakayama, Y., Tanishiro, Y., and Takayanagi, K. (2011). Quantitative annular dark-field STEM images of a silicon crystal using a large-angle convergent electron probe with a 300-kV cold field-emission gun. *J. Electron. Microsc.* 60, 109–116. <https://doi.org/10.1093/jmicro/dfq084>.
24. Zhang, M., Ming, Y., Zeng, R.G., and Ding, Z.J. (2015). Calculation of Bohmian quantum trajectories for STEM. *J. Microsc.* 260, 200–207. <https://doi.org/10.1111/jmi.12283>.
25. Ruan, Z., Zeng, R.G., Ming, Y., Zhang, M., Da, B., Mao, S.F., and Ding, Z.J. (2015). Quantum-trajectory Monte Carlo method for study of electron-crystal interaction in STEM. *Phys. Chem. Chem. Phys.* 17, 17628–17637. <https://doi.org/10.1039/c5cp02300a>.
26. Dwyer, C., and Etheridge, J. (2003). Scattering of Å-scale electron probes in silicon. *Ultramicroscopy* 96, 343–360. [https://doi.org/10.1016/s0304-3991\(03\)00100-1](https://doi.org/10.1016/s0304-3991(03)00100-1).
27. Dwyer, C. (2005). Multislice theory of fast electron scattering incorporating atomic inner-shell ionization. *Ultramicroscopy* 104, 141–151. <https://doi.org/10.1016/j.ultramic.2005.03.005>.
28. Bethe, H. (1928). Theorie der Beugung von Elektronen an Kristallen. *Ann. Phys.* 392, 55–129.
29. Cowley, J.M. (1995). *Diffraction Physics* (Elsevier).
30. Brigham, E.O., and Morrow, R.E. (1967). The fast Fourier transform. *IEEE Spectr.* 4, 63–70. <https://doi.org/10.1109/mspec.1967.5217220>.
31. Coppersmith, D. (1994). An approximate Fourier transform useful in quantum factoring. Preprint at: arXiv. <https://doi.org/10.48550/arXiv.quant-ph/0201067>.
32. Nielsen, M.A., and Chuang, I.L. (2010). *Quantum Computation and Quantum Information* (Cambridge university press).
33. Zindorf, B., and Bose, S. (2024). Efficient Implementation of Multi-Controlled Quantum Gates. Preprint at: arXiv. <https://doi.org/10.48550/arXiv.2404.02279>.
34. Welch, J., Greenbaum, D., Mostame, S., and Aspuru-Guzik, A. (2014). Efficient quantum circuits for diagonal unitaries without ancillas. *New J. Phys.* 16, 033040. <https://doi.org/10.1088/1367-2630/16/3/033040>.
35. Araujo, I.F., Park, D.K., Ludermir, T.B., Oliveira, W.R., Petruccione, F., and da Silva, A.J. (2023). Configurable sublinear circuits for quantum state preparation. *Quantum Inf. Process.* 22, 123. <https://doi.org/10.1007/s11128-023-03869-7>.
36. Gleinig, N., and Hoefer, T. (2021). An Efficient Algorithm for Sparse Quantum State Preparation. In 58th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA (IEEE). <https://doi.org/10.1109/dac18074.2021.9586240>.
37. O'Donnell, R., and Wright, J. (2016). Efficient quantum tomography. In Proceedings of the forty-eighth annual ACM symposium on Theory of Computing, D. Wichs and Y. Mansour, eds. (Association for Computing Machinery). <https://doi.org/10.1145/237814.237866>.
38. Walsh, J.L. (1923). A closed set of normal orthogonal functions. *Am. J. Math.* 45, 5–24. <https://doi.org/10.2307/2387224>.
39. Gilbert, E.N. (1958). Gray codes and paths on the N-cube. *Bell Syst. Tech. J.* 37, 815–826. <https://doi.org/10.1002/j.1538-7305.1958.tb03887.x>.
40. Hamming, R.W. (1950). Error detecting and error correcting codes. *Bell Syst. Tech. J.* 29, 147–160. <https://doi.org/10.1002/j.1538-7305.1950.tb00463.x>.
41. Dou, M., Zou, T., Fang, Y., Wang, J., Zhao, D., Yu, L., Chen, B., Guo, W., Li, Y., Chen, Z., et al. (2022). QPanda: high-performance quantum computing framework for multiple application scenarios arXiv. Preprint at: arXiv. <https://doi.org/10.48550/arXiv.2212.14201>.
42. Chen, Z.Y., and Guo, G.P. (2019). Qrunes: High-level language for quantum-classical hybrid programming. Preprint at: arXiv. <https://doi.org/10.48550/arXiv.1901.08340>.
43. Peng, L.M. (1999). Electron atomic scattering factors and scattering potentials of crystals. *Micron* 30, 625–648. [https://doi.org/10.1016/s0968-4328\(99\)00033-5](https://doi.org/10.1016/s0968-4328(99)00033-5).

**STAR★METHODS****KEY RESOURCES TABLE**

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Software and algorithms		
Python	Open source	<a href="https://www.python.org/">https://www.python.org/</a>
PyQPanda	OriginQC	<a href="https://github.com/OriginQ/QPanda-2">https://github.com/OriginQ/QPanda-2</a>

**EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS**

Our study does not use experimental models typical in the life sciences.

**METHOD DETAILS**

We use a classical supercomputer to simulate a quantum circuit with a Python package “pyqpanda” at the level of qubit and quantum gate, and run the quantum algorithm on the virtual quantum circuit. We calculate the atomic potential field for each slice of the crystal cell and compute the potential operators and kinetic operators. After that we simulate the quantum circuit to execute the quantum algorithm.

In the quantum algorithm, we initialize the quantum state to represent the initial state of the electron, then we construct a quantum circuit to perform iterations and obtain results. The quantum circuit incorporates quantum Fourier transforms, inverse quantum Fourier transforms, and phase-shifting quantum circuits corresponding to potential operators and kinetic operators. To evaluate improvements, we compared two versions of the quantum algorithm: before and after optimization. In the previous version, we construct the phase-shifting quantum circuit directly based on the operators, whereas in the later improved version of the quantum algorithm, we perform the Walsh transform on the operators to obtain the Walsh coefficients of each operator, apply different truncation thresholds, and construct the improved version of the phase-shifting quantum circuits based on the Walsh coefficients. We use Python to perform the entire simulation and to analyze and visualize the results.

**QUANTIFICATION AND STATISTICAL ANALYSIS**

We use the number of Walsh terms to measure the complexity of the phase-shifting quantum circuit, thereby reflecting the speedup due to truncation, as the number of Walsh terms directly determines the number of quantum gates.

We use Python to calculate the truncation errors for different truncation thresholds. The truncation error is defined by [Equation 16](#).