

A quantum algorithm of multislice method

Y.C. Wang^a, Y. Sun^{c,*}, Z.J. Ding^{a,b,*}

^a Department of Physics, University of Science and Technology of China, Hefei, Anhui 230026, PR China

^b Hefei National Research Center for Physical Sciences at the Microscale, University of Science and Technology of China, Hefei, Anhui 230026, PR China

^c Department of Physics, Xiamen University, Xiamen, Fujian 361005, PR China

ARTICLE INFO

Keywords:

Quantum algorithm
Multislice method
Electron scattering and diffraction
Quantum computing
Quantum simulation

ABSTRACT

Multislice method is widely used to simulate the elastic interaction of an electron beam with matter in electron microscopy. In this paper, we propose a quantum algorithm of multislice method. With the use of quantum circuit this quantum version of multislice method costs less storage space and less runtime than the classical algorithm by taking advantage of the feature of quantum computing. We have used a classical supercomputer to simulate a quantum circuit, and successfully carried out the quantum multislice computation and obtained the expected result on a virtual quantum circuit, which proves the feasibility and correctness of the quantum algorithm. This work demonstrates the potential application prospect of quantum computing to the research of computational physics as a good example.

Introduction

Quantum computing means doing computation with quantum physical systems that obey quantum mechanical laws. The idea of quantum computing was proposed a long time ago by Feynman [1,2]. In recent years, there has been a booming interest in quantum computing due to great progresses made on the quantum computer [3–9]. Especially superconducting quantum circuit becomes one of the most promising way to achieve large-scale fault-tolerant universal quantum computer [10,11]. For example, in 2019 a superconducting quantum processor ‘Sycamore’ with 53 qubits developed by Google has been claimed to have achieved quantum supremacy in random quantum circuit sampling [4]. In 2021, a USTC team has developed a faster quantum processor ‘Zuchongzhi’ with 66 qubits, and claimed to have established an unambiguous quantum computational advantage [7]. These works indicate that we are getting closer and closer to large-scale quantum computing.

Theoretically, by harnessing features of quantum system like superposition and entanglement, quantum computing can gain a strong speedup over classical computing, even solve problems that cannot be done classically in a proper time. An examples is the Shor’s algorithm for the integer factorization problem [12], which can be solved exponentially faster than the best-known classical algorithm can. However, to perform Shor’s algorithm practically one still needs a much larger

number of qubits than we can achieve nowadays [13]. Other applications of quantum computing include the database searching [14], quantum simulation [15], quantum machine learning [16] and etc. [17–19].

In this paper, we propose a quantum algorithm of multislice method, which applies quantum computing to the computational modelling of the elastic interaction of an electron beam with a matter to gain advantages in time complexity and storage memory requirement over a classical algorithm. The algorithm only requires a small number of qubits; thus the work provides a strong proof for the practical application value of quantum computing in the field of computational physics.

Multislice method is a widely-used method to simulate the propagation of the electron wave function in the three-dimensional atomic potential field, especially for the elastic interaction of the high-energy electron beam with sample in transmission electron microscopy. The original multislice method was proposed by Cowley et al. in 1957 [20–22]. Goodman and Moodie established a numerical procedure in 1974 [23]. In 1977, Ishizuka and Uyeda proposed a multislice formula derived directly from the integral form of the Schrodinger equation [24], with which a fast Fourier transform [25] is useful to perform the convolution integral. Later on, multislice method was extended and improved further [26–29]. It has been successfully used for the simulation of TEM images [30,31], STEM images [32–34], Bohmian quantum trajectories [35,36], electron energy loss spectra (EELS) [37,38]. It was

* Corresponding authors at: Department of Physics, Xiamen University, Xiamen, Fujian 361005, PR China (Y. Sun). Department of Physics, University of Science and Technology of China, Hefei, Anhui 230026, PR China (Z.J. Ding).

E-mail addresses: yangsun@xmu.edu.cn (Y. Sun), zjding@ustc.edu.cn (Z.J. Ding).

also used to perform a full-space analysis applied to the simulation of mechanical interaction of electron vortex beam with a nanoparticle [39].

Another theoretical method to simulate electron propagation is Bloch wave method [40–43]. Bloch wave method is more suitable for perfect crystals; but for crystal with defects or noncrystal with more complex structure this method has limitations. On the contrary, multislice method can in principle handle well all kinds of complex non-periodic potential field. But, as object size in simulation increases, the computation cost of multislice method will be accordingly expanded in a power law.

The essence of multislice method is to iteratively solve the Schrödinger equation in two-dimensional potential field. Benenti and Strini have proposed a method to solve Schrödinger equation in one-dimensional simple resolvable potential field with quantum algorithm based on quantum circuit [44]. Here we extend this idea to the multislice method and consider two-dimensional complex atomic potential field situation. In this way, we can completely use quantum circuit built up by basic quantum gates to perform layer-by-layer iteration of the electron wave function in the multi-layer method and to derive the desired quantum algorithm of multilayer method. Through this method, one can obtain the probability density distribution for the electron wave propagation in a three-dimensional sample as well as the electron diffraction pattern. Moreover, we illustrate that this quantum algorithm would be better than the classical algorithm in terms of both time complexity and storage space requirement.

Theory

Multislice method

In multislice method, the sample is divided into evenly spaced slices perpendicularly to the incident direction of electron beam, and the potential field is approximately compressed and projected to the interface between slices. The wave function is approximated as propagating freely within a slice, and is only affected by a two-dimensional potential field as it passes through each interface.

The detailed theoretical derivation of the multislice method starts from the Schrödinger equation in the three-dimensional space. Assume the incident direction of electron beam to be along the z -axis. The wave function of electrons, $\psi_f(x, y, z)$, propagation in the electrostatic potential field, $V(x, y, z)$, satisfies the following time-independent Schrödinger equation,

$$\left[-\frac{\hbar^2 \nabla^2}{2m} - eV(x, y, z) \right] \psi_f(x, y, z) = E_k \psi_f(x, y, z) \quad (1)$$

where \hbar is the reduced Planck constant, m the mass of an electron, e the elementary charge. E_k is the kinetic energy of an electron and relates to the electron wavelength λ by,

$$E_k = \frac{\hbar^2}{2m\lambda} \quad (2)$$

At the accelerating voltage U of an electron beam the kinetic energy of the electrons is

$$E_k = eU \quad (3)$$

For fast electrons moving in the atomic potential field, the effect of the potential to the electron momentum can also be regarded as a small perturbation. Thus, the electron wave function can be written as,

$$\psi_f(x, y, z) = \psi(x, y, z) \exp\left(\frac{2\pi iz}{\lambda}\right) \quad (4)$$

where $\psi(x, y, z)$ represents the small perturbation and the term $\exp(2\pi iz/\lambda)$ represents a plane wave. Using Eqs. (2) & (4), one can rewrite Eq. (1) as

$$-\frac{\hbar^2}{2m} \left[\nabla_{xy}^2 + \frac{\partial^2}{\partial z^2} + \frac{4\pi i}{\lambda} \frac{\partial}{\partial z} + \frac{2meV(x, y, z)}{\hbar^2} \right] \psi(x, y, z) = 0 \quad (5)$$

where ∇_{xy}^2 is the sum of second derivatives with respect to coordinates x and y .

At very high energy, electron wave length λ is very small; then we have

$$\left| \frac{\partial^2 \psi}{\partial z^2} \right| \ll \left| \frac{4\pi i}{\lambda} \frac{\partial \psi}{\partial z} \right| \quad (6)$$

Eq. (5) can then be approximated as

$$-\frac{\hbar^2}{2m} \left[\nabla_{xy}^2 + \frac{4\pi i}{\lambda} \frac{\partial}{\partial z} + \frac{2meV(x, y, z)}{\hbar^2} \right] \psi(x, y, z) = 0 \quad (7)$$

Dropping the $\frac{\partial^2 \psi}{\partial z^2}$ term is confirmed to be a good approximation for high energy electrons and has been used widely in electron diffraction calculations [45]. More specifically, this term will only be of importance when ψ varies significantly over a distance equal to the electron wavelength, leading to a small quantitative error on radiiuses of HOLZ (Higher order Laue zone) which is typically of order 1 % for 200 keV electrons and smaller at higher voltages [46].

Eq. (7) gives the derivative with respect to z , which is

$$\begin{aligned} \frac{\partial \psi(x, y, z)}{\partial z} &= \left[\frac{i\lambda}{4\pi} \nabla_{xy}^2 + i \frac{2me\lambda}{4\pi\hbar^2} V(x, y, z) \right] \psi(x, y, z) \\ &= \left[\frac{i\lambda}{4\pi} \nabla_{xy}^2 + i\sigma V(x, y, z) \right] \psi(x, y, z). \end{aligned} \quad (8)$$

where $\sigma = \frac{me\lambda}{2\pi\hbar^2}$. Taking the integral of Eq. (8), we can obtain

$$\psi(x, y, z + \Delta z) = \exp \left[\int_z^{z+\Delta z} \left(\frac{i\lambda}{4\pi} \nabla_{xy}^2 + i\sigma V(x, y, z') \right) dz' \right] \psi(x, y, z) \quad (9)$$

which shows how wave function changes when z changes by Δz . When Δz is very small, Eq. (9) is reduced to

$$\psi(x, y, z + \Delta z) = \exp \left(\frac{i\lambda}{4\pi} \Delta z \nabla_{xy}^2 + i\sigma V_{\Delta z}(x, y, z) \Delta z \right) \psi(x, y, z) \quad (10)$$

where $V_{\Delta z}(x, y, z)$ is the average potential between z and $z + \Delta z$, given by

$$V_{\Delta z}(x, y, z) = \frac{1}{\Delta z} \int_z^{z+\Delta z} V(x, y, z') dz' \quad (11)$$

Ishizuka et al. proposed that the maximum slice thickness Δz should be $\sim kd^2$ in order to get a stable result close to the limiting value, where k is the wavenumber of the incident electrons and d is the distance over which the potential does not change by an appreciable fraction [24].

∇_{xy}^2 has following relationship with \hat{p}_{xy} :

$$\nabla_{xy}^2 = \left(\frac{i\hat{p}_{xy}}{\hbar} \right)^2 = -\frac{\hat{p}_{xy}^2}{\hbar^2} \quad (12)$$

where \hat{p}_{xy} is the momentum operator in the xy -plane. And the momentum p_{xy} is proportional to the space Q_{xy} :

$$p_{xy} = 2\pi\hbar Q_{xy} \quad (13)$$

Substituting Eqs. (12)-(13) into Eq. (10), we obtain

$$\psi(x, y, z + \Delta z) = \exp \left(-i\pi\Delta z\lambda |Q_{xy}|^2 + i\sigma V_{\Delta z}(x, y, z) \Delta z \right) \psi(x, y, z) \quad (14)$$

where $\exp(-i\pi\Delta z\lambda |Q_{xy}|^2)$ is called propagator.

More specifically, considering the evolution of the electron wave function from the slice t to the slice $t + 1$, we have

$$\psi_{t+1}(\mathbf{r}) = \exp(-i\pi d\lambda|\mathbf{Q}|^2) \exp(i\sigma dV_t(\mathbf{r})) \psi_t(\mathbf{r}) \quad (15)$$

where d is the separation between slices, $\psi_t(\mathbf{r})$ is the wave function at slice t , $\mathbf{r} = (x, y)$ represents the space coordinates in the xy -plane, \mathbf{Q} represents the space frequency (Q_x, Q_y) in the plane. The potential energy term $\exp(i\sigma dV_t(\mathbf{r}))$ is diagonal in the coordinate representation, so it can act directly on the wave function $\psi_t(\mathbf{r})$ in the coordinate representation to obtain

$$\psi'_t(\mathbf{r}) = \exp(i\sigma dV_t(\mathbf{r})) \psi_t(\mathbf{r}) \quad (16)$$

The propagator $\exp(-i\pi d\lambda|\mathbf{Q}|^2)$, on the contrary, is not diagonal in the coordinate representation, but it is diagonal in the momentum representation. Therefore, one needs firstly to use Fourier transform (FT) to transform the wave function in the coordinate representation to the momentum representation:

$$\varphi'_t(\mathbf{Q}) = \text{FT}[\psi'_t(\mathbf{r})] \quad (17)$$

Then the propagator can act directly on the wave function to obtain

$$\varphi'_{t+1}(\mathbf{Q}) = \exp(-i\pi d\lambda|\mathbf{Q}|^2) \varphi'_t(\mathbf{Q}) \quad (18)$$

Finally, one can use inverse Fourier transform (iFT) to transform the wave function of the slice $t+1$ back to the coordinate representation to obtain the evolving result of this slice,

$$\psi_{t+1}(\mathbf{r}) = \text{iFT}[\varphi'_{t+1}(\mathbf{Q})] \quad (19)$$

which will be used for the next slice calculation. In this way, from Eqs. (16)-(19), we complete the evolution for one slice as schematically shown by Fig. 1 [39]. For more slices, we only need to repeat this process.

In calculation, we use a $N \times N$ matrix to store the wave function of one slice discretely. For transformation between the coordinate representation and the momentum representation, we just need to apply discrete Fourier transform (DFT) or inverse discrete Fourier transform (iDFT) to this matrix. The range that is covered by the matrix in the momentum space is related to the range in the coordinate space. For instance, the range in coordinate space is $(-\frac{L_x}{2}, \frac{L_x}{2})$ in the x -direction, the total length is L_x . Then in the momentum space, the interval between sample point is $\frac{1}{L_x}$ in the x -direction, and the range is $(-\frac{N}{2} \left(\frac{1}{L_x} \right), (\frac{N}{2} - 1) \left(\frac{1}{L_x} \right))$.

When the energy of electrons is high, we need to consider the relativistic effect. The relativistic total energy is,

$$E = E_0 + E_k = m_0 c^2 + eU \quad (20)$$

where m_0 is the rest mass of electron, c is the speed of light in vacuum. According to the mass-energy equation,

$$E = mc^2 \quad (21)$$

one obtains

$$m = m_0 + eU/c^2 \quad (22)$$

where m is the relativistic mass of electron, which will be used in the calculation. In addition, we have the following energy-momentum relation:

$$E^2 = p^2 c^2 + m_0^2 c^4 \quad (23)$$

where p is the momentum. Using Eq. (20) and Eq. (23) we have

$$p = \frac{1}{c} \sqrt{2E_0 E_k + E_k^2} \quad (24)$$

Using de-Borglie wavelength relation, $\lambda = h/p$, we have

$$\lambda = \frac{hc}{\sqrt{2E_0 E_k + E_k^2}} \quad (25)$$

Quantum computing

Quantum computing refers to computation making use of physical systems with quantum properties. By taking advantage of the superposition and entanglement of quantum systems, it is possible to bring exponential optimization of the time or space complexity of classical algorithms. In 1982, Feynman proposed that when using a classical computer to simulate a quantum system, the consumption of computational resources would increase exponentially with the number of particles and degrees of freedom of the system being simulated, and that it is more appropriate and natural to use an actual quantum system to simulate quantum phenomena, whereby the concept of quantum computing was introduced [1]. The algorithms for quantum computing are called quantum algorithms.

Quantum circuit is a widely used model of quantum computing. It was systematically discussed in 1986 by Feynman with an early version of the quantum circuit notation [2]. A quantum circuit is similar to a classical logic circuit. In classical logic circuits, bits are the basic storage unit. Each classical bit has two physical state representing 0 and 1. The input information is encoded in a number of bits at the input end and then passed through some logic gates in turn to obtain the desired output result. Similarly, in a quantum circuit, qubits are the basic storage unit. A qubit generally also has two orthogonal physical state representing 0 and 1, which are called basis states and can be noted as $|0\rangle$ and $|1\rangle$ using Dirac notation. Unlike classical bits which can only be in one of two states, quantum bits, however, can be in a quantum superposition state of two basis states, whose quantum states can be expressed as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (26)$$

where α and β are two complex numbers, the complex amplitudes,

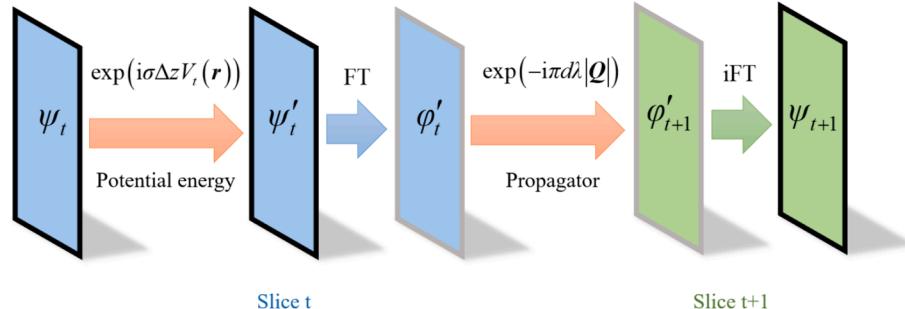


Fig. 1. The flowchart of the wave function evolution process between two slices in the multislice method. The black frame represents the coordinate representation, while the gray one represents the momentum representation.

satisfying

$$|\alpha|^2 + |\beta|^2 = 1 \quad (27)$$

If we perform a measurement on this qubit, it will randomly collapse to one of the basis states. The probability of getting 0 is $|\alpha|^2$ and the probability of getting 1 is $|\beta|^2$. A classical register consisting of n bits has 2^n possible states totally, similarly a quantum register consisting of n qubits has 2^n basis states. We use $|k\rangle$ to represent the k th basis state, where $k \in \{0, 1, \dots, 2^n - 1\}$, then the quantum state of the whole quantum register can be written as,

$$|\psi\rangle = \sum_{k=0}^{2^n-1} c_k |k\rangle \quad (28)$$

where

$$\sum_{k=0}^{2^n-1} |c_k|^2 = 1 \quad (29)$$

Similar to classical logic gate, a quantum logic gate can have a specific operation on the data stored in quantum register, the effect of which is multiplying a transformation matrix on the state vector. According to quantum mechanics, the transformation must be reversible and unitary, which means that a quantum logic gate must have equal number of input qubits and output qubits.

For example, for a typical single-bit quantum gate, the Hardmard gate, the corresponding transformation matrix is

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (30)$$

Then if we apply it on a qubit whose initial state is $|0\rangle$ and state vector is $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$, we can calculate the output quantum state:

$$H|0\rangle = H \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \quad (31)$$

By specific assembly of qubits and quantum logic gates, and proper initialization of the input quantum state to encode the required input for computation, one can implement various quantum algorithms using quantum circuits. Next, we specifically introduce an important quantum algorithm, quantum Fourier transform (QFT), and the quantum circuit used for implementation.

Quantum Fourier transform

QFT [47,48] is a quantum algorithm analogous to the classical discrete Fourier transform (DFT) or an inverse discrete Fourier transform (iDFT) applied to a quantum state vector according to different definitions. QFT is very important and is used in many famous quantum algorithms as a sub algorithm, such as Shor's algorithm [12] and quantum phase estimation algorithm [49]. It is also a critical sub algorithm in our present quantum algorithm.

First, we consider only one-dimensional case. The definition of sign of the phase factor in QFT is various, and we will use a common definition here. Considering the length of a quantum state vector is $N = 2^n$, and the input state is $|x\rangle = \sum_{k=0}^{N-1} x_k |k\rangle$, then the QFT will map it to $|y\rangle = \sum_{k=0}^{N-1} y_k |k\rangle$, where

$$y_k = \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} x_l e^{2\pi i k l / N} \quad (32)$$

In this definition, QFT equals to act iDFT to a quantum state vector. And for inverse quantum Fourier transform (iQFT),

$$y_k = \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} x_l e^{-2\pi i k l / N} \quad (33)$$

it equals to act DFT to a quantum state vector. Note that when the input state $|x\rangle$ is one of the basis states $|k\rangle$, according Eq. (32), QFT maps it as follows:

$$|k\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} e^{2\pi i k l / N} |l\rangle \quad (34)$$

This relationship holds for all basis states, and any quantum state is a linear superposition of basis states, so Eq. (34) can also be used as the definition of QFT.

Similarly, iQFT maps the basis state as follows:

$$|k\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} e^{-2\pi i k l / N} |l\rangle \quad (35)$$

We use the quantum circuit in Fig. 2 to implement QFT [48]. This quantum circuit consists of two types of quantum gates, Hardmard gates and controlled phase-shift gates. Here H represents the Hardmard gate described above, and R_m represents a controlled phase-shift gate with a specific phase shift, corresponding to the transformation matrix,

$$R_m = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i / 2^m} \end{bmatrix} \quad (36)$$

As for iQFT, since all quantum gates are corresponding to invertible transformations, it needs only to reverse the order of all quantum gates in the QFT quantum circuit to obtain the iQFT quantum circuit.

According to the diagram of QFT circuit, only $O(n^2) = O((\log_2 N)^2)$ quantum gates are needed for QFT with modulo N . In comparison, for fast Fourier transform [25] (FFT), one of the fastest classical algorithms of DFT, $O(2^n n) = O(N \log_2 N)$ unit operations are needed for the same modulo. Besides, as the modulo increases, the storage space required for a classical computer to store the quantum state increases as $O(N)$, while for a quantum computer the number of qubits required is $O(n) = O(\log_2 N)$. Therefore, QFT is much better than classical FFT exponentially both in terms of time complexity and required storage space.

To perform a QFT on present-day quantum circuits in a fault-tolerant manner, we need additional error-correcting process. The resource required for a fault-tolerant QFT at a certain fidelity was discussed by Goto [50], while we will focus on our quantum algorithm in this paper and assume that we have fault-tolerant qubits and quantum gates, regardless of the limitation of near-term quantum devices. In the following we will transform the classical multislice method into a quantum multislice algorithm with quantum circuits, which will be able to run on a universal quantum computer. In this process, DFT will be replaced by QFT, and the rest parts of the algorithm will also be reconstructed accordingly.

Quantum multislice algorithm

We use a two-dimensional amplitude encoding to store each slice of the wave function. The two-dimensional wave function will be discretized into a $N \times N$ complex matrix where $N = 2^n$; then there are totally $N^2 = 2^{2n}$ elements in the square matrix. Each element corresponds to the amplitude of the wave function at a coordinate in the real spatial plane. The wave function is expressed as $\psi(k_x, k_y)$, where k_x and k_y are index numbers of the matrix ($k_x, k_y \in \{0, 1, \dots, N-1\}$). To store this matrix, we need a quantum register with $2n$ qubits, which have 2^{2n} basis states.

We use the amplitude of each basis state to represent the amplitude of each coordinate point. More specifically, we divide a quantum register into two parts, each with n qubits. According to the binary code, the lower n bits represent the x -coordinate and the higher n bits represent the y -coordinate. For instance, for $n = 3$, the basis state $|011001_2\rangle = |011_2\rangle |001_2\rangle = |3\rangle |1\rangle = |k_y\rangle |k_x\rangle$, then we let its amplitude be $\psi(1, 3)$, where $|3\rangle |1\rangle$ is the abbreviation of $|3\rangle \otimes |1\rangle$. The electron quantum state

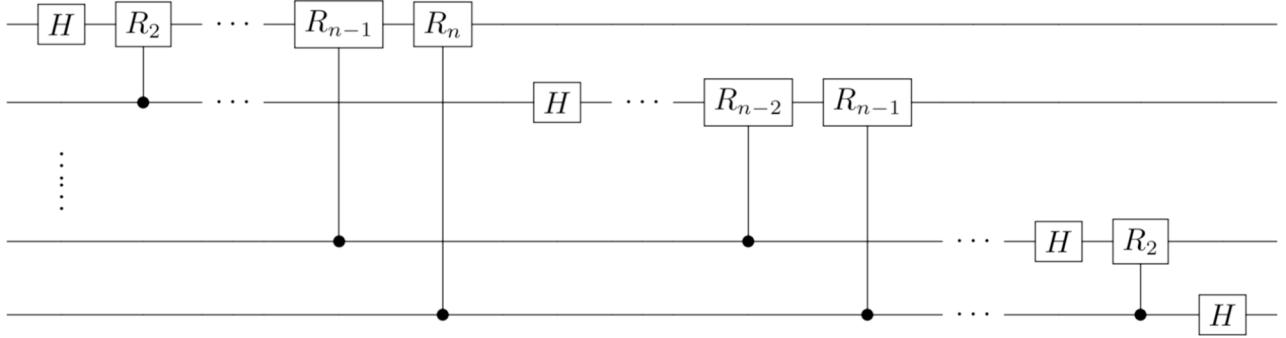


Fig. 2. The diagram of QFT quantum circuit.

must satisfy the normalization condition, that is

$$\sum_{k_x, k_y} |\psi(k_x, k_y)|^2 = 1 \quad (37)$$

So the quantum state of the quantum register will naturally satisfy the normalization condition at the same time.

The range of the simulated space can be handled in different ways. For one slice, if we approximate the sample as a perfect crystal, we can only simulate the cross-sectional size of only one cell and obtain the whole slice of wave function by symmetry. For an irregular potential field, we need to simulate a larger spatial range, which of course will increase the computational cost. This paper aims mainly to demonstrate the algorithm, so we consider to simulate only the cross-sectional size of one crystal cell. Assume that the crystal is a cubic crystal with lattice constant a_0 , and the electron beam is incident along the z axis, then the spatial resolution in the x - y plane is a_0/N , and the spatial resolution in the z direction is equal to the slice spacing d of the multislice method. And if the wave function is transformed into momentum representation with spatial frequency \mathbf{Q} as coordinate by FT, the resolution of spatial frequency will be $1/a_0$, the simulated range will be $\left(-\frac{N}{2}\left(\frac{1}{a_0}\right), \left(\frac{N}{2}-1\right)\left(\frac{1}{a_0}\right)\right)$.

To begin this quantum algorithm, we initialize the wave function. The initial state of the wave function is determined by the incident electron beam. Considering the simplest case by assuming plane wave electron beam, then the amplitudes at all coordinates of the initial state are equal, i.e.

$$\psi(k_x, k_y) = \sqrt{1/N^2} \quad (38)$$

To obtain this uniform initial state one only needs an all-zero state and one layer of H gates acting on each qubit, totally $O(n) = O(\log_2 N)$ gates, which is very efficient. Other arbitrary forms of initial quantum states can be obtained by a sublinear quantum circuit with complexity no more than $O(N)$ [51]. In certain circumstances where the sample structure is nonperiodic and when one wishes to extend the simulated area, sometimes far more than the electron beam incident area, the initial state of electron will be a sparse state. In these situations, an efficient algorithm for sparse quantum state preparation can be used to optimize the complexity to $O(sn) = O(s \log_2 N)$, where s is the number of nonzero coefficients [52]. In this way one can extend the simulated area without wasting linear overhead on initiating grid points where there is no electron beam incident.

After the initialization, we allow the wave function to propagate layer by layer. Using Eqs. (16)-(19), we can write the evolution process of one slice as

$$\psi_{t+1}(x, y) = F^{-1} e^{-i\pi d \lambda |Q_x^2 + Q_y^2|} F e^{i\pi d V_t(x, y)} \psi_t(x, y) \quad (39)$$

where F represents FT and F^{-1} represents iFT. We will fully use quantum algorithm based on quantum circuit to complete this process.

First, we need to achieve the FT and the iFT using quantum circuit. We have already discussed the simple one-dimensional case; but here we have to consider the two-dimensional FT. Our specific method is to apply a n -qubit iQFT to the n qubits encoding x -coordinate and another n -qubit iQFT to the n qubits encoding y -coordinate at the same time. In this way we can achieve a two-dimensional iQFT which is equal to applying a two-dimensional DFT to the $N \times N$ square state matrix. The proof is as follows.

Assuming the input quantum state to be $|k_y\rangle|k_x\rangle$, when we apply iQFT to both higher n qubits and lower n qubits separately, according to Eq. (35), the quantum state will change as follows,

$$\begin{aligned} |k_y\rangle|k_x\rangle &\rightarrow \frac{1}{N} \left(\sum_{l_y=0}^{N-1} e^{-2\pi i k_y l_y / N} |l_y\rangle \right) \otimes \left(\sum_{l_x=0}^{N-1} e^{-2\pi i k_x l_x / N} |l_x\rangle \right) \\ &= \frac{1}{N} \sum_{l_y=0}^{N-1} \sum_{l_x=0}^{N-1} e^{-2\pi i (k_x l_x + k_y l_y) / N} |l_y\rangle|l_x\rangle. \end{aligned} \quad (40)$$

Then if the input quantum state is $\psi = \sum_{k_y=0}^{N-1} \sum_{k_x=0}^{N-1} f(k_y, k_x) |k_y\rangle|k_x\rangle$, according to Eq. (40), it will change as follows,

$$\begin{aligned} \psi &\rightarrow \frac{1}{N} \sum_{k_y=0}^{N-1} \sum_{k_x=0}^{N-1} f(k_y, k_x) \sum_{l_y=0}^{N-1} \sum_{l_x=0}^{N-1} e^{-2\pi i (k_x l_x + k_y l_y) / N} |l_y\rangle|l_x\rangle \\ &= \sum_{l_y=0}^{N-1} \sum_{l_x=0}^{N-1} \left[\frac{1}{N} \sum_{k_y=0}^{N-1} \sum_{k_x=0}^{N-1} f(k_y, k_x) e^{-2\pi i (k_x l_x + k_y l_y) / N} \right] |l_y\rangle|l_x\rangle \\ &= \sum_{l_y=0}^{N-1} \sum_{l_x=0}^{N-1} F(l_y, l_x) |l_y\rangle|l_x\rangle, \end{aligned} \quad (41)$$

where

$$\begin{aligned} F(l_y, l_x) &= \frac{1}{N} \sum_{k_y=0}^{N-1} \sum_{k_x=0}^{N-1} e^{-2\pi i (k_x l_x + k_y l_y) / N} f(k_y, k_x) \\ &= \frac{1}{N} \sum_{\mathbf{k}} e^{-2\pi i (\mathbf{k} \cdot \mathbf{l}) / N} f(\mathbf{k}), \end{aligned} \quad (42)$$

which is exactly in the form of a two-dimensional QFT.

Similarly, we can perform a two-dimensional iDFT on the quantum state matrix by applying two n -qubit QFT to both lower n qubits and higher n qubits separately at the same time.

Apart from FT and iFT, both of other two terms in Eq. (39) are diagonal phase operation satisfying the following form,

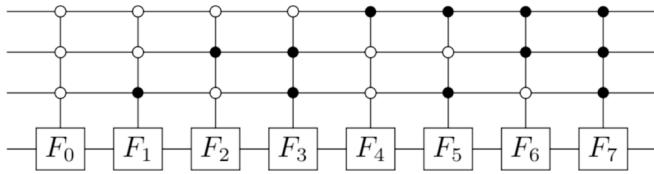
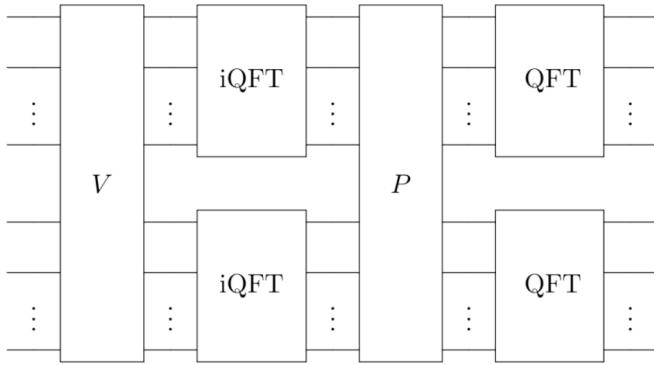
Fig. 3. The phase-shifting quantum circuit for $n = 2$.

Fig. 4. Diagram of the quantum circuit required for the evolution of one slice of electron wave function.

$$|x\rangle \rightarrow e^{if(x)}|x\rangle \quad (43)$$

which is just an application of a phase shift $f(x)$ to a basis state $|x\rangle$. It does not matter whether $|x\rangle$ is decoded in one dimension or two dimensions. We present here one basic quantum circuit that can achieve this operation. There are other more complex but more efficient sub-algorithms for this part, which will not be discussed here [53].

Assuming that the quantum register has $2n$ qubits, then the above operation can be achieved by 2^{2n-1} generalized controlled phase-shift gates [44]. These gates apply a single-qubit transformation F_k to the target qubit which is the lowest qubit, if and only if the other $2n-1$ control qubits are in the quantum state $|k\rangle$, where $k \in \{0, 1, \dots, 2^{2n-1} - 1\}$, and F_k operation corresponds to the matrix,

$$F_k = \begin{bmatrix} e^{if(2k)} & 0 \\ 0 & e^{if(2k+1)} \end{bmatrix} \quad (44)$$

For example, the quantum circuit for $n = 2$ is as shown by Fig. 3.

In this way, with three fundamental quantum circuit modules, QFT, iQFT, and diagonal phase-shifting operation, one can achieve the evolution of one slice of the electron wave function completely using quantum algorithm.

Fig. 4 shows a diagram of the complete quantum circuit required for the evolution of one slice of the electron wave function. The simulation of the electron wave function in three-dimensional space can be achieved by cycling this process. The electron wave function can be decoded from the final output quantum state, which includes the position information and momentum information of the electron at different depths. In practical quantum computing, how to efficiently extract information from the quantum state is also a problem worthy of study. The most basic approach is doing a standard projective measurement after each run and repeat the quantum simulation. The measured results will form a distribution which matches with the probability distribution of the electron state, while each run will output a possible location or momentum of the electron, which can be considered as a single sampling. Although the number of repetitions might add an extra factor to the overall complexity and cancel out the time advantage, that is not always the case. The number of repetitions needed depends on the shape of the distribution and the statistical error we can accept. We present a sampling demonstration of the momentum distribution to reconstruct

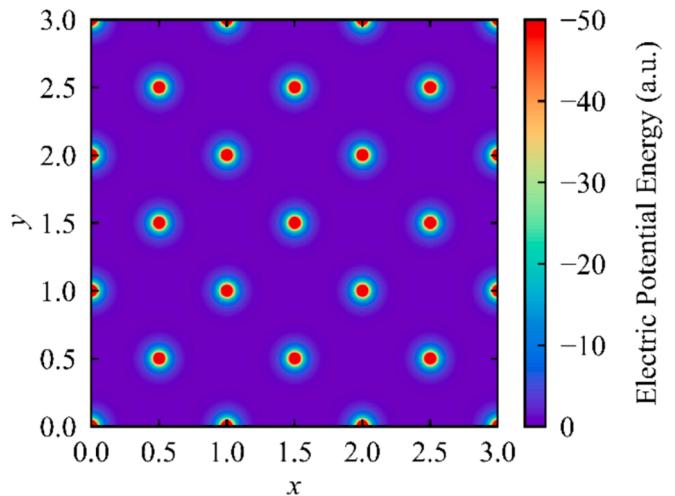


Fig. 5. The contour map of the electric potential field of Au crystal in 280 K, using lattice constant as unit size length.

the electron diffraction pattern with a limited number of samplings, as a brief example of information extraction. We can expect the momentum distribution to have a number of sharp peaks corresponding to diffraction peaks, and thus samples will land on peaks with a high probability; therefore, fewer repetitions are needed to form the pattern. Advanced extraction of full quantum state would involve quantum state tomography [54] which will not be discussed in this paper. Besides extracting information directly from the output quantum state, one can also use the output quantum state in other ways to improve the utilization. For example, 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 which might be useful for further simulations.

Results and discussion

At present there is no practical programmable quantum computer that has enough number of qubits with small enough error rate for us to actually carry out this algorithm, therefore, we use a classical computer

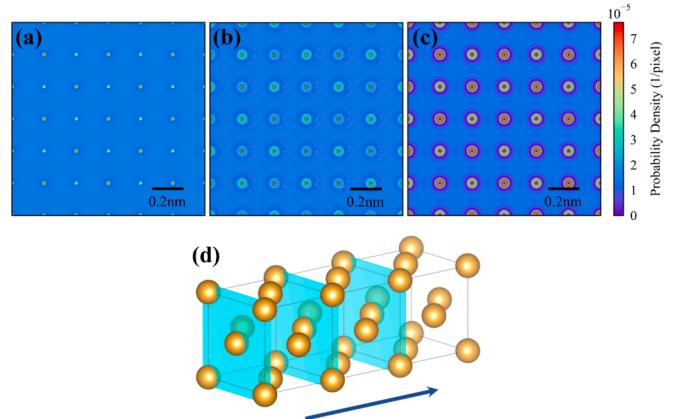


Fig. 6. Contour maps of electron probability density distributions on the cross sectional planes at different depths: (a) $t = 96$, (b) $t = 256$, (c) $t = 512$, where t represents the layer number of the iteration in multislice method, for plane wave incident electrons at 100 keV. Each figure includes 3×3 unit cells with 256×256 pixels per unit cell, and each unit cell is divided by 32 slices in the electron propagating direction, satisfying the limitation of the maximum slice thickness. (d) The diagram of cross sections perpendicular to the electron propagating direction. The arrow indicates the incident direction of the electron.

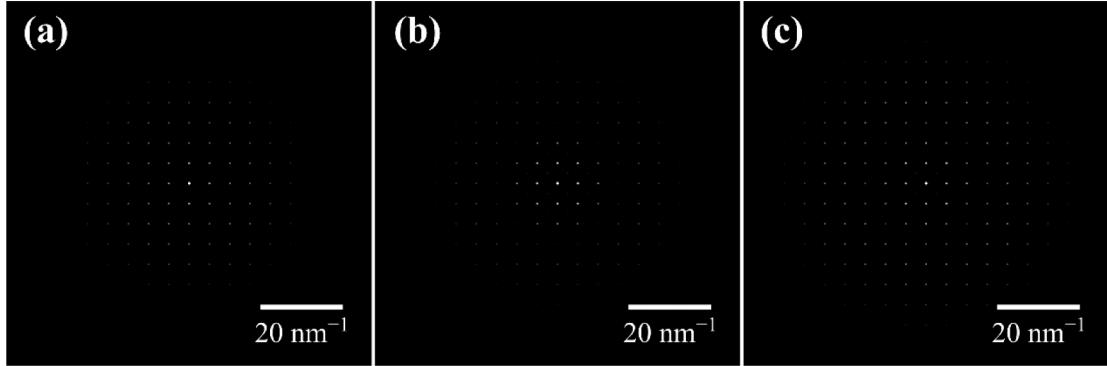


Fig. 7. The electron spatial frequency distributions on the cross sectional planes at different depths: (a) $t = 160$, (b) $t = 320$, (c) $t = 640$, where t represents the layer number of the iteration in multislice method, for plane wave incident electrons at 100 keV.

to simulate the quantum circuit to verify the feasibility and validity of this quantum algorithm. Unlike running classical multislice algorithm on a supercomputer, we first simulate a quantum circuit at the level of qubit and quantum gate using a python package “pyqpanda” [55,56], and then run the quantum algorithm on the virtual quantum circuit. In this method, we simulated electron wave function and electron diffraction pattern for electrons at different energies incident into a thick Au specimen under different parameters.

At first we use the following formula to calculate the atomic potential field in the crystal [57]:

$$\varphi(\mathbf{r}) = \frac{\hbar^2}{2\pi m_0 e} \sum_i a_i \left(\frac{4\pi}{b_i + B} \right)^{3/2} \exp \left(-\frac{4\pi^2 r^2}{b_i + B} \right) \quad (45)$$

where a_i and b_i ($i = 1, 2, 3, 4$) are fitting parameters, B is the Debye-Waller factor that depends on temperature. Fig. 5 shows a cross section of the potential field of an Au crystal at an atom plane at 280 K, including 3×3 unit cells.

Using the potential field in Eq. (45) and the above mentioned quantum circuits in Fig. 4 to iterate in cycle, we have calculated the electron probability density distribution on the cross sectional plane perpendicular to the electron propagating direction at the electron depth. Fig. 6 shows the typical density distributions at several depths. Here we have used 16 qubits for the simulation, i.e. $n = 8, N = 2^8 = 256$, and there are 256×256 pixels per unit cell.

By transforming the output probability density distributions into the momentum representation, one can obtain the electron momentum distribution on a cross sectional plane at the corresponding electron depth. Fig. 7 illustrates the calculated electron spatial frequency distributions, which are proportional to electron momentum distributions and can be used to represent the electron far-field diffraction patterns according to quantum mechanical principle and diffraction theory [41].

Also, one can obtain similarly the side cross sectional electron probability density distribution as shown by Fig. 8 on a plane parallel to the electron incident direction. Here one can see the electron channeling effect because the incident electron beam is parallel to the atom columns

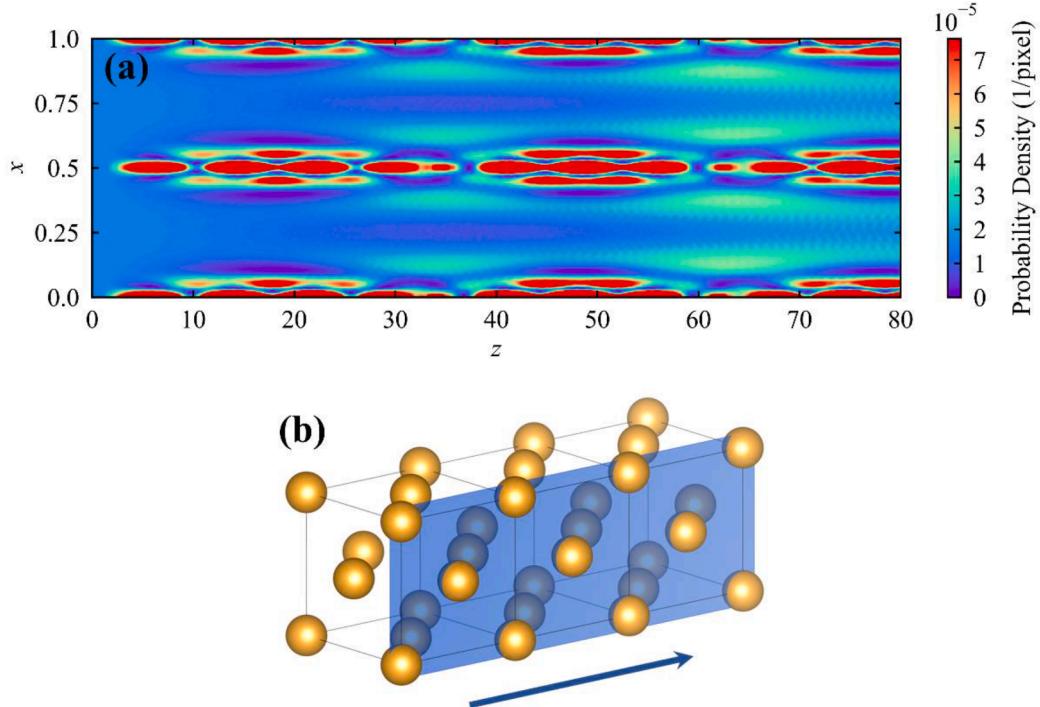


Fig. 8. (a) The side cross sectional electron probability density distribution for plane wave incident electrons at 100 keV, using lattice constant as the unit length. Each unit cell is divided by 32 slices in z-direction. We used $n = 16$ qubits for the simulation. (b) The diagram of a side cross section plane parallel to the electron propagating direction. The arrow indicates the incident direction of the electron.

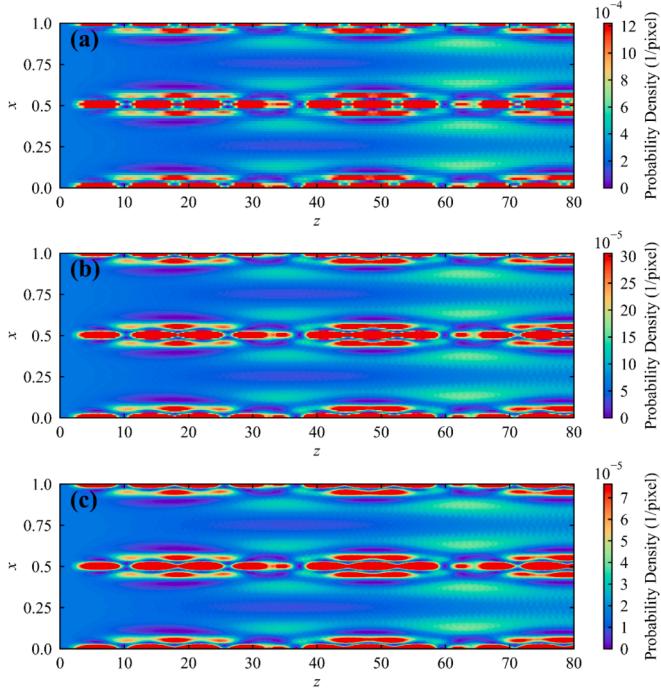


Fig. 9. The calculated side cross sectional electron probability density distributions for plane wave incident electrons at 100 keV by using different numbers of qubits: (a) $n = 12$, (b) $n = 14$, (c) $n = 16$. The numbers of pixels per unit cell are: (a) 64×64 , (b) 128×128 , (c) 256×256 .

[58]. The electron wave is focused at periodic distances on the atom columns. The period is called extinction distance.

In order to compare the effect of different quantum computation parameters on the calculation results, we have performed calculations of the side cross sectional electron probability density distribution by using various different parameters. Fig. 9 compares the calculation results by using different numbers of qubits. Using different n could lead to different spatial resolutions or computational accuracies. It can be seen that there are slight differences in the calculation results as we change the numbers of qubits, but the distribution feature remains basically stable, and using 16 quantum bits is enough to get a considerable computation accuracy.

We have also performed a classical multislice method using the same parameters as in Fig. 9(a). Results are compared in Fig. 10. The result obtained by quantum algorithm is completely in agreement with that

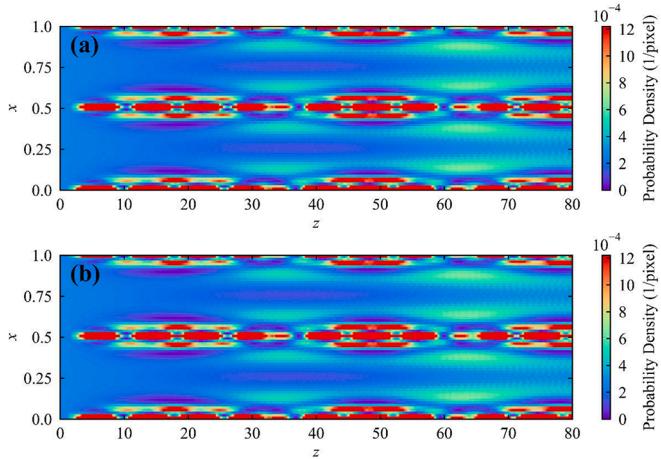


Fig. 10. (a) The result of quantum algorithm. (b) The result of classical algorithm.

obtained by the classical multislice method, which proves the feasibility and validity of the quantum algorithm.

Fig. 11 shows the calculated side cross sectional electron probability density distributions for different electron incident energies. It can be seen that the extinction distance increases as the incident energy increasing.

As commonly used, here we measure the computation complexity of the quantum algorithm by the number of quantum gates in the quantum circuit, which can be defined as the size of the quantum circuit. We use the inner function of "pyqpanda" to count the quantum gates, where all types of quantum gates with one target qubit are considered equally, regardless of the number of control qubits. For a comparison, we measure the computation complexity of the corresponding classical algorithm by the number of gates or unit operations.

The FT part is what we are most interested in, because in the classical algorithm, FFT is the most complex part and the most time-consuming part. Theoretically, the QFT can achieve a huge complexity advantage over the FFT. Assume n qubits are used in the quantum algorithm, then the corresponding spatial scale is $N = 2^n$. As mentioned previously, the theoretical complexity of the FFT is $O(2^n n) = O(N \log_2 N)$, while that of the QFT is $O(n^2) = O((\log_2 N)^2)$. We count the actual number of quantum gates used in our simulated quantum circuits and compare it with the theoretical complexity of the classical FFT algorithm, as Fig. 12. It can be seen that as the simulating scale increasing the quantum algorithm has much slower increase of computation complexity compared to the classical algorithm.

However, for the rest of the quantum algorithm which is mainly the parts for the phase operations corresponding to terms $e^{-i\pi d_x^2 |Q_x^2 + Q_y^2|}$ and $e^{i\pi d V_t(x,y)}$ in Eq. (39), we have not yet optimized sufficiently and used the most basic scheme; therefore, the computation complexity of these parts is comparable to that of the classical algorithm, both being $O(N)$, without significant advantage. Nevertheless, the overall complexity of the classical algorithm is $O(N \log_2 N) + O(N) = O(N \log_2 N)$, while that of the quantum algorithm is $O((\log_2 N)^2) + O(N) = O(N)$. In other words, we have obtained an exponential speedup on the most time-consuming part of the algorithm, so even though the complexity of the rest part remains unchanged, for the overall complexity of the algorithm, the present quantum algorithm still has a certain computation advantage over the classical algorithm. We counted the number of quantum gates used in our quantum algorithm for the evolution of one slice of electron wave function, and compared it with the number of unit operations of the classical algorithm. The results are shown in Fig. 13. One can see that the quantum algorithm requires fewer logic gates than the classical algorithm as the simulating scale increasing.

It is worth to clarify that we use a large number of multi-controlled quantum gates in the current version of phase-shifting circuits, which could be difficult to be implemented directly on near-term quantum processors. In this situation, multi-controlled quantum gates would be decomposed into one- and two-qubit gates during compilation, adding a new factor to the complexity of the phase-shifting part, $O(\log_2 N)$, if we use the decomposition method [59] as an example. This would nullify the complexity advantage of the quantum algorithm. However, we are currently working on optimizing the phase-shifting circuits [53] and have already achieved some preliminary results. In our future work, we should be able to replace all multi-controlled gates with a comparable number of two-qubit gates to solve this potential problem. After the replacement, there will still be room for optimization on the phase-shifting circuits, and the quantum algorithm can achieve a greater complexity advantage. This is also a feasible direction for our follow-up research.

As mentioned in the previous chapter, we present a sampling demonstration to show the possibility of extracting non-trivial information with a controllable number of samplings. We repeat the quantum simulation and each run is followed by a projective measurement in the

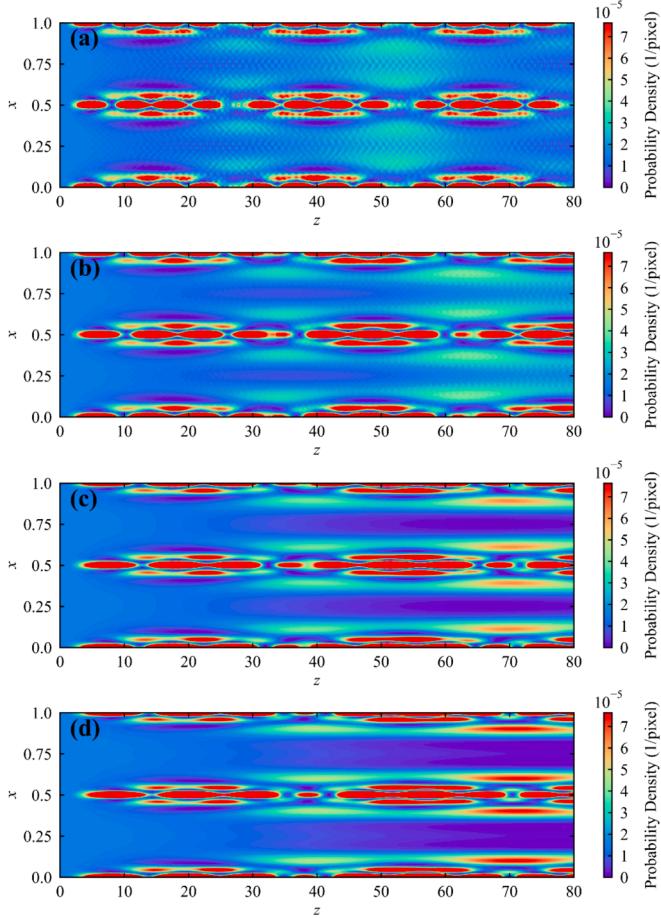


Fig. 11. The calculated side cross sectional electron probability density distributions for plane wave incident electrons at different energies: (a) 50 keV, (b) 100 keV, (c) 200 keV, (d) 300 keV, by using 16 qubits.

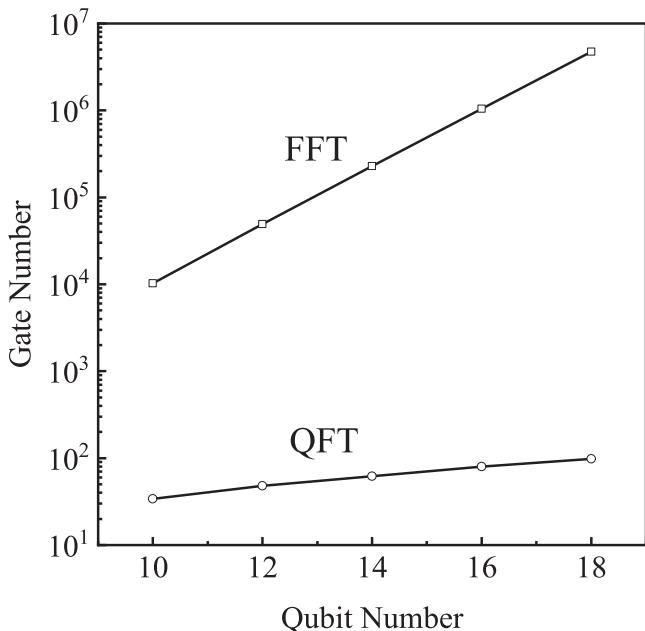


Fig. 12. The complexity comparison for QFT part in the simulated quantum circuit and FFT in theory, where the vertical axis is logarithmic.

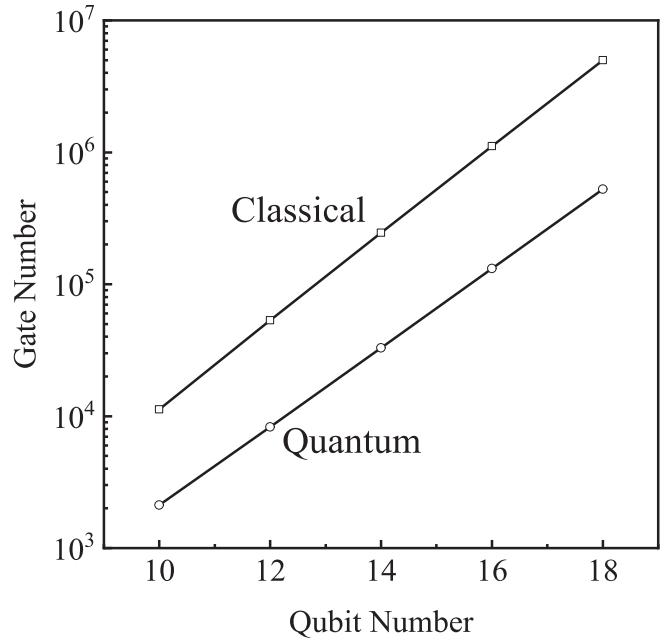


Fig. 13. The overall complexity comparison for the quantum algorithm and the classical algorithm.

momentum representation. Each measurement outputs one of the basis states which corresponds to a grid point; then we accumulate and overlap the samples to form a pattern. The results are shown in Fig. 14.

As we expected, the samples land on the diffraction peaks with a high probability and form the pattern easily. The pattern appears when the sampling number reaches 500 and becomes clear as the sampling number increases to 5000, which is much smaller than the grid number, $2^{16} = 65536$ in this case. Moreover, the required sampling number will not depend on N since the diffraction peaks remain fixed as N increasing. Thus, in this scenario, information extraction only adds a constant factor to the overall complexity and does not cancel out the quantum advantage.

Conclusion

In this paper, we propose a quantum algorithm for the multislice method, can be implemented by using quantum circuits to simulate the propagation of electron wave function in a three-dimensional atomic potential field, and thus obtain the spatial probability density distribution of electrons as well as the diffraction pattern. Although the existing state-of-the-art quantum computing hardware is still insufficient for performing this quantum algorithm, we have verified the feasibility and validity of the quantum algorithm of the multislice method by simulating the quantum circuit with a classical supercomputer. Compared with the classical algorithm, our quantum algorithm of the multislice method has certain computation advantages in terms of wave function storage space requirement and time complexity, especially for the FT part, which can reduce the time complexity from $O(2^n n)$ to $O(n^2)$. For the computation complexity of other parts in the quantum algorithm there is still a room for further optimization, which is worthy of further research. Although only the simplest case of plane wave and periodic potential field has been considered for performing the calculation in this paper, but our quantum algorithm has the theoretical ability to extend to more complex incident waves and atomic potential fields.

With this quantum algorithm of the multislice method, we present a very practical prospect of possible application and show the importance of quantum computing in the physics research. It is believed that with the further development of quantum computing hardware, the quantum algorithm of multislice method will have a chance to be practically

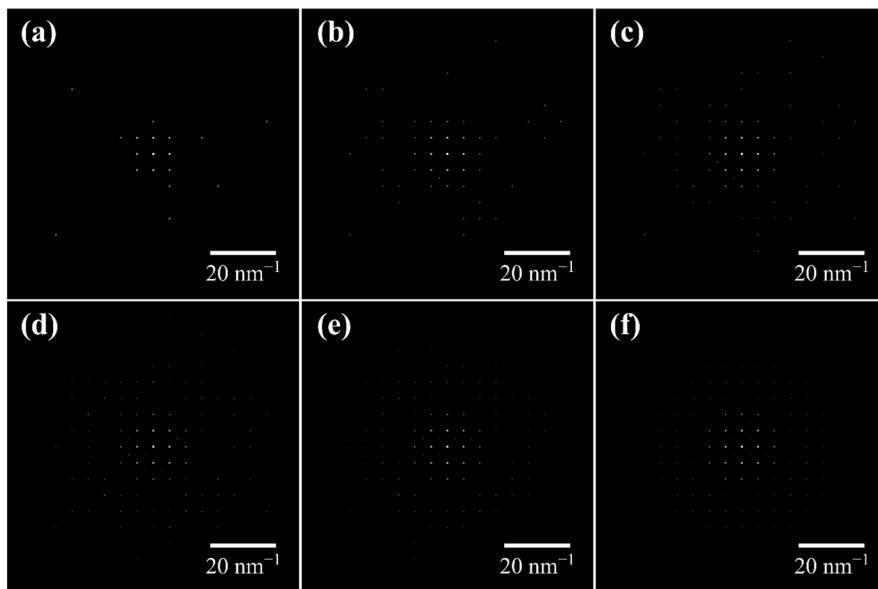


Fig. 14. (a)-(e) The sampling results. The sampling numbers are: (a) 100, (b) 500, (c) 1000, (d) 2000, (e) 5000. (f) The full result of diffraction simulation, as in Fig. 7(b).

performed on a quantum computer.

CRediT authorship contribution statement

Y.C. Wang: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis. **Y. Sun:** Funding acquisition, Software. **Z.J. Ding:** Writing – review & editing, Supervision, Project administration, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work is supported by the Chinese Education Ministry through “111 Project 2.0” (BP0719016), the Natural Science Foundation of Xiamen (3502Z202371007), the Fundamental Research Funds for the Central Universities (20720230014) and National Natural Science Foundation of China (42374108). We thank Prof. H.M. Li and the supercomputing center of USTC for the support of parallel computing. We acknowledge the support by TianHe Qingsuo Project-special fund project from the National Supercomputer Center in Tianjin.

References

- [1] Feynman RP. *Inter J Theor Phys* 1982;21:467.
- [2] Feynman RP. *Found Phys* 1986;16:507.
- [3] Wright K, Beck KM, Debnath S, Amini JM, Nam Y, Grzesiak N, et al. *Nat Comm* 2019;10:5464.
- [4] Arute F, Arya K, Babbush R, Bacon D, Bardin JC, Barends R, et al. *Nature* 2019;574:505.
- [5] Zhong HS, Wang H, Deng YH, Chen MC, Peng LC, Luo YH, et al. *Science* 2020;370:1460.
- [6] Zhong HS, Deng YH, Qin J, Wang H, Chen MC, Peng LC, et al. *Phys Rev Lett* 2021;127:180502.
- [7] Wu YL, Bao WS, Cao SR, Chen FS, Chen MC, Chen XW, et al. *Phys Rev Lett* 2021;127:180501.
- [8] Madsen LS, Laudenbach F, Askarani MF, Rortais F, Vincent T, Bulmer JFF, et al. *Nature* 2022;606:75.
- [9] Philips SGJ, Amitonov SV, de Snoo SL, Russ M, Kalhor N, Volk C, et al. *Nature* 2022;609:919.
- [10] M. Kjaergaard, M. E. Schwartz, J. Braumüller, P. Krantz, J. I. J. Wang, S. Gustavsson, and W. D. Oliver, *Annu. Rev. Condens. Ma.* P. 11, 369 (2020).
- [11] Huang HL, Wu DC, Fan DJ, Zhu XB. *Sci China-Inform Sci* 2020;63:180501.
- [12] P. W. Shor, in: 35th Annual Symposium on Foundations of Computer Science (Santa Fe, 1994), pp. 124.
- [13] Gouzien E, Sangouard N. *Phys Rev Lett* 2021;127:140503.
- [14] L. K. Grover, Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, 212 (1996).
- [15] Georgescu IM, Ashhab S, Nori F. *Rev Mod Phys* 2014;86:153.
- [16] Biamonte J, Wittek P, Pancotti N, Rebentrost P, Wiebe N, Lloyd S. *Nature* 2017;549:195.
- [17] Harrow AW, Hassidim A, Lloyd S. *Phys Rev Lett* 2009;103:150502.
- [18] Peruzzo A, McClean J, Shadbolt P, Yung MH, Zhou XQ, Love PJ, et al. *Nature Com* 2014;5:4213.
- [19] E. Farhi, J. Goldstone, and S. Gutmann, arXiv:1411.4028 (2014).
- [20] Cowley JM, Moodie AF. *Acta Crystal* 1957;10:609.
- [21] Cowley JM, Moodie AF. *Acta Crystal* 1959;12:353.
- [22] Cowley JM, Moodie AF. *Acta Crystal* 1959;12:360.
- [23] Goodman P, Moodie AF. *Acta Crystal A* 1974;30:289.
- [24] Ishizuka K, Uyeda N. *Acta Crystal A* 1977;33:740.
- [25] Brigham EO, Morrow RE. *IEEE Spectr* 1967;4:63.
- [26] van Dyck D. *Phys Status Solidi* a 1979;52:283.
- [27] Van Dyck D. *J Microsc* 1980;119:141.
- [28] Ishizuka K. *Acta Crystal A* 1982;38:773.
- [29] Kilaas R, O’Keefe MA, Krishnan KM. *Ultramicrosc* 1987;21:47.
- [30] Van Dyck D. *Adv Imag Elect Phys* 1985;65:295.
- [31] O’Keefe MA, Kilaas R. *Users Guide to NCEMSS Version 1994*;2.
- [32] Ishizuka K. *J Elect Microsc* 2001;50:291.
- [33] Ishizuka K. *Ultramicrosc* 2002;90:71.
- [34] Kim S, Oshima Y, Sawada H, Kaneyama T, Kondo Y, Takeguchi M, et al. *J Elect Microsc* 2011;60:109.
- [35] Zhang M, Ming Y, Zeng RG, Ding ZJ. *J Microsc* 2015;260:200.
- [36] Ruan Z, Zeng RG, Ming Y, Zhang M, Da B, Mao SF, et al. *Phys Chem Chem Phys* 2015;17:17628.
- [37] Dwyer C, Etheridge J. *Ultramicrosc* 2003;96:343.
- [38] Dwyer C. *Ultramicrosc* 2005;104:141.
- [39] Shi T, Ding Z. *Ultramicrosc* 2022;238:113551.
- [40] Bethe H. *Annalen der Physik* 1928;392:55.
- [41] Cowley JM. *Diffraction physics*. Amsterdam: Elsevier; 1995.
- [42] Reimer L, Kohl H. *Transmission Electron Microscopy*. Berlin: Springer; 2008.
- [43] Cheng L, Ming Y, Ding ZJ. *New J Phys* 2018;20:113004.
- [44] Benenti G, Strini G. *Am J Phys* 2008;76:657.
- [45] Howie A, Basinski ZS. *Philos Mag* 1968;17:1039.
- [46] Bird DM. *J Electron Microsc* Tech 1989;13:77.
- [47] D. Coppersmith, arXiv:quant-ph/0201067v1 (1994).
- [48] Nielsen MA, Chuang IL. *Quantum computation and quantum information*. Cambridge: Cambridge University Press; 2010.

- [49] A. Y. Kitaev, arXiv:quant-ph/9511026 (1995).
- [50] Goto H. Phys Rev A 2014;90:052318.
- [51] Araujo IF, Park DK, Ludermir TB, Oliveira WR, Petruccione F, da Silva AJ. Quantum Inf Process 2023;22:123.
- [52] Gleinig N, Hoefer T. in: 58th ACM/IEEE Design Automation Conference (DAC). San Francisco: CA; 2021. p. 433.
- [53] Welch J, Greenbaum D, Mostame S, Aspuru-Guzik A. New J Phys 2014;16:033040.
- [54] R. O'Donnell and J. Wright, in: Proceedings of the forty-eighth annual ACM symposium on Theory of Computing (Cambridge, MA, 2016), pp. 899.
- [55] M. Dou, T. Zou, Y. Fang, J. Wang, D. Zhao, L. Yu, B. Chen, W. Guo, Y. Li, Z. Chen, and G. Guo, arXiv preprint arXiv:2212.14201 (2022).
- [56] Z. Y. Chen and G. P. Guo, arXiv:1901.08340 [quant-ph] (2019).
- [57] Peng LM. Micron 1999;30:625.
- [58] Van Aert S, Geuens P, Van Dyck D, Kisielowski C, Jinschek JR. Ultramicrosc 2007; 107:551.
- [59] B. Zindorf and S. Bose, arXiv preprint arXiv:2404.02279 (2024).