Circuit implementation of the globally optimized HHL algorithm and experiments on qiskit

Meng Zhang
Xidian Univ
Lihua Dong (lih_dong@mail.xidian.edu.cn)
Xidian Univ
Yong Zeng
Xidian Univ
Ning Cao
Xidian Univ

Article

Keywords:

Posted Date: April 8th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1505699/v1

Circuit implementation of the globally optimized HHL algorithm and experiments on qiskit

Meng Zhang[1], Lihua Dong[1*], Yong Zeng[2], Ning Cao[1]

(1. School of Communication Engineering, Xidian Univ., Xi'an 710071, China;
(2. School of Network and Information Security, Xidian Univ., Xi'an 710071, China)

Abstract: In 2019, Yonghae Lee et al. combined the circuit implementation of the HHL quantum algorithm with a classical computer, and designed a hybrid HHL quantum circuit optimization algorithm to reduce experimental errors caused by decoherence and so on. However, the optimization is achieved only in the auxiliary quantum coding phase, and no quantum resource reduction is done on the quantum phase estimation and inverse quantum phase estimation stages. At the same time, the circuit optimization illustration on the 2th-order linear equation system just has the result and no specific process. In this paper, based on the idea of the hybrid HHL algorithm and the general quantum circuit implementation framework of HHL, a global optimization HHL algorithm is proposed. The feasibility of the global optimization HHL algorithm is verified by IBM's qiskit. The detail circuit optimization illustrations on the 4th-order linear equations show that the global optimization HHL algorithm can effectively reduce quantum resources without losing the fidelity of the results. Thus the global optimization HHL algorithm can further avoid some result errors than the existing implementation methods.

Introduction

Quantum computing is an operation method that follows the operating laws of quantum mechanics. Compared with traditional classical computing, quantum computing achieves an exponential speedup on some problems. For example, the Shor quantum algorithm [1]is famous for factoring large integers in polynomial time. Grover algorithm [2]achieves an exponential speedup in searching data. While the HHL algorithm achieves an exponential acceleration [3].

However, in the process of mapping quantum algorithms to quantum circuits, due to the limitations of current technology, the errors of quantum gates, experimental errors and decoherence will introduce errors in the experimental process [4]. Therefore, it is necessary to reduce the number of gates and the overall running time of the algorithm. In view of this, the realization and optimization of quantum circuits has attracted the attention of various fields.

Among them, in terms of quantum circuit implementation and optimization of cryptographic algorithms, B-Langenberg et al. proposed the quantum circuit implementation and optimization of AES cryptographic algorithm [7], and many scholars have made further optimizations on it [8].

In circuit synthesis of quantum algorithms, T Monz et al. implemented the use of Shor's algorithm to factorize numbers 15 [10]7 qubits and 4 cache qubits with efficient control. Z Diao et al. proposed the quantum circuit composition of [11]. Markus Grassl successfully used the Grover algorithm to realize the exhaustive key search for [12]. The quantum circuit implementation and optimization of the HHL algorithm also attracted a lot of attention. Among them, by introducing the variable time amplitude amplification algorithm into the HHL algorithm, Ambainis et al. reduced the number of repeated runs required to obtain the correct answer, thereby reducing the running time of the algorithm [13], but did not give a specific quantum circuit implementation. Yudong Cao et al. proposed a general HHL algorithm quantum circuit implementation [14]. Yonghae Lee et al. gave a hybrid HHL quantum algorithm **Error! Reference source not found.**, which effectively reduces the quantum gate resources used in the auxiliary qubit rotation part of the HHL algorithm. Compared with the general HHL algorithm quantum circuit implementation proposed by Cao et al., the hybrid HHL algorithm uses smaller quantum resources in the auxiliary qubit rotation stage. However, the optimization has only limited to the auxiliary quantum encoding stage, and no quantum resource can be reduced on the quantum phase estimation and inverse quantum phase estimation stages. Meanwhile, in the circuit optimization verification, there is only optimization result on the 2th-order linear equation system and no specific realization process is given.

In view of this, inspired by the idea of combining with classical computers and the general HHL quantum circuit implementation, we provide a global optimization quantum circuit diagram of the HHL algorithm to further reducing the number of quantum gates, thus further avoiding some result errors caused by quantum gate errors. The experimental results show that our globally optimized HHL algorithm effectively reduces the consumption of quantum resources without losing the fidelity of the results.

Basic definitions

- ***k-fixed*** **[6]**: Suppose λ_j, $j = 1, \dots, l$, is all non-zero eigenvalues of the Hermitian matrix A, b_k^j is the *k-th* bit of the binary representation of the eigenvalues $\lambda_j, j = 1,\dots, l$, defined $\bar{m}_k$, $k \in N$, as follows

$$\bar{m}_k = \frac{1}{l}\left(\sum_{j=1}^{l} b_k^j\right) \qquad (1)$$

Hermitian matrix A is said to be *k -fixed* if it is $\bar{m}_k$ fixed at 0 or 1.

- ***n-represented*** **[21]**: Denoted λ as the eigenvalue of the matrix, if it can be represented λ by no more than n binary number, it is called *n-represented*
- **fidelity *f* [3]**: The fidelity index is defined as

$$f = |\langle x(t)|x_e(t)\rangle|^2 \qquad (2)$$

Among them $|x(t)\rangle$ is the state obtained by the evolution of the initial state $|x(0)\rangle$ without interference, and is the state obtained by the evolution of the $|x_e(t)\rangle$ initial state $|x(0)\rangle$ in the case of interference, that is, the fidelity can be understood as the inner product of the theoretical value and the experimental value.

For example, assuming that the theoretical normalized solution of a $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ linear system of equations is, and the actual normalized solution is $\begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$, then the fidelity is $f = \sqrt{x_1 y_1 + x_2 y_2}$.

HHL algorithm [3]

Quantum Circuit Implementation of HHL Algorithm: The quantum circuit implementation of the HHL algorithm requires the use of three quantum registers, denoted as $Ancilla$, $Re\,g.C$ and $Re\,g.B$, where

- $Ancilla$ is used to store auxiliary qubits;
- $Re\,g.C$ is used to store the binary representation of the eigenvalues of the coefficient matrix A;
- $Re\,g.B$ is used to store the vector solution of a system of linear equations when the measurement of the contents of the $Ancilla$quantum register is 1.
- Initially, all three quantum registers are set to $|0\rangle$ state.

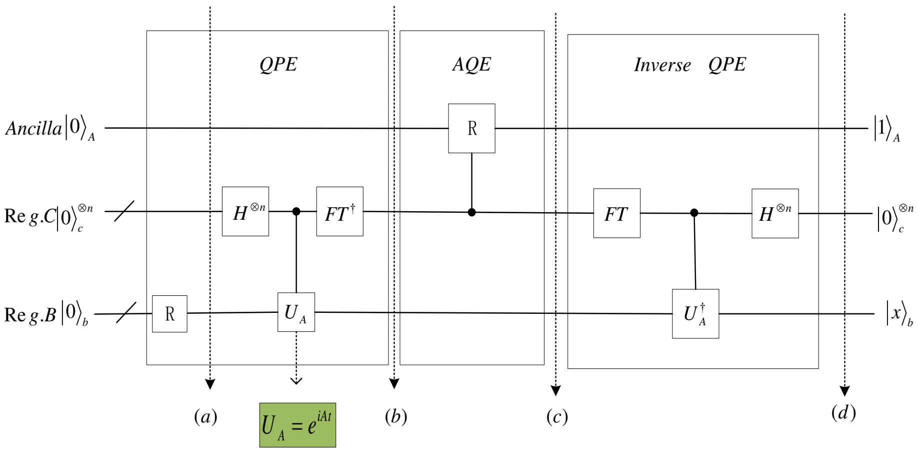


Figure 1 Circuit overview diagram of HHL algorithm

As shown in **Figure 1** Circuit overview diagram of HHL algorithm, the quantum circuit implementation process of the HHL algorithm is mainly composed of three stages, namely: quantum phase estimation (Quantum Phase Estimation, QPE), auxiliary quantum encoding (Ancilla quantum encoding, AQE) and inverse quantum phase estimation (Inverse Quantum Phase Estimation, Inverse QPE) of which

a) The essence of QPE is to estimate the eigenvalues of the coefficient matrix A. It is the core subroutine of the HHL algorithm and the core routine of many quantum algorithms.

Firstly, the conversion from the ground state to the mixed state suitable for storing the phase is completed, and the quantum state after the conversion is completed is

$$|0\rangle_A \otimes \left(H^{\otimes n} \otimes I\right)|0\rangle \otimes |0\rangle_b = |0\rangle_A \otimes \frac{1}{\sqrt{2^n}} \sum_{x_0 x_1 \ldots x_{n-1} \in \{0,1\}^n} |x_0 x_1 \ldots x_{n-1}\rangle \otimes |b\rangle \quad x_i \in \{0,1\}. \tag{3}$$

Secondly, use n control U gates $C - U_A^j$ $j = 2^0, \ldots 2^{n-1}$ to addition of phase $e^{2\pi i \theta_j}$, $j = 2^0, \ldots 2^{n-1}$, to probability amplitude, which θ_j is a real number in the interval(0,1).

$$|0\rangle_A \otimes \frac{(|0\rangle+e^{2\pi i\theta_0}|1\rangle)(|0\rangle+e^{2\pi i\theta_1}|1\rangle)...(|0\rangle+e^{2\pi i\theta_j}|1\rangle)}{2^{n/2}} \otimes |b\rangle \quad (4)$$

Finally, the information on the probability amplitude of the qubit is stored in the quantum state, and the final quantum state is

$$|0\rangle_A \otimes \frac{1}{\sqrt{2^n}} \sum_{i=0}^{n-1} |\theta_i\rangle \otimes |b\rangle \quad (5)$$

In the entire HHL algorithm, the unitary operator $U = U_A = e^{iAt} = \sum_{j=0}^{N-1} e^{i\lambda_j t} |u_j\rangle\langle u_j|$, $|b\rangle = \sum_{j=0}^{N-1} b_j |u_j\rangle$, where t is a constant that can be set by itself. Therefore, in the entire HHL algorithm, the state of time, that is, the moment in **Figure 1** Circuit overview diagram of HHL algorithm is (b):

$$|0\rangle_A \otimes \sum_{j=0}^{N-1} b_j |\lambda_j\rangle_{nl} \otimes |u_j\rangle \quad (6)$$

$$|u\rangle = \sum_{j=0}^{N-1} b_j |u_j\rangle, \qquad |\lambda_j\rangle_{nl} = \frac{1}{\sqrt{2^n}} \sum_{i=0}^{n-1} |\theta_i\rangle$$

Here $|\lambda_j\rangle_{nl}$ is the n-bit binary representation of "λ_j".

b) In the AQE stage, a controlled rotation operation on auxiliary qubits is performed as follows:

$$|0\rangle_A \otimes |\lambda_j\rangle_{nl} \to \left(\sqrt{1-\frac{c^2}{\lambda_j^2}}|0\rangle_A + \frac{c}{\lambda_j}|1\rangle_A\right) \otimes |\lambda_j\rangle_{nl} \quad (7)$$

Here c is a normalizing constant.

After the AQE part, the state of the system at moment **Figure 1** Circuit overview diagram of HHL algorithm(c) is:

$$\sum_{j=0}^{N-1} b_j |\lambda_j\rangle_{nl} |u_j\rangle \left(\sqrt{1-\frac{c^2}{\lambda_j^2}}|0\rangle_A + \frac{c}{\lambda_j}|1\rangle_A\right) \quad (8)$$

c) Inverse QPE stage is the inverse operation of the QPE stage. After the QPE stage, the $|\lambda_j\rangle$ in the superposition state in the $Reg.C$ register will become $|0\rangle$, at this time, the state of the entire quantum system is **Figure 1** Circuit overview diagram of HHL algorithm(d):

$$\sum_{j=0}^{N-1} b_j |0\rangle_{nl} |u_j\rangle \left(\sqrt{1-\frac{c^2}{\lambda_j^2}}|0\rangle_A + \frac{c}{\lambda_j}|1\rangle_A\right) \quad (9)$$

After the above three stages are completed, $Reg.C$ all qubits in the quantum register are set to the $|0\rangle$ state, and the auxiliary qubit is measured on the Z-axis. If the measurement result is $|1\rangle_A$, that is, the output state of $|1\rangle$ *Ancilla* is, then the system successfully solves the system of equations and obtains the normalized solution of the system of equations is as follows:

$$|x\rangle = \left(\sqrt{\frac{1}{\sum_{j=0}^{N-1}|b_j|^2/|\lambda_j|^2}}\right) \sum_{j=0}^{N-1} \frac{b_j}{\lambda_j} |0\rangle_{nl} |u_j\rangle. \quad (10)$$

General HHL Algorithm Quantum Circuit Implementation: In 2012, Yudong Cao et al. proposed an efficient and general quantum circuit design implementation [14]. In this design, the Group Leader Optimization Algorithm was used to find the circuit decomposition of the [16]. Then simply multiply the offset angles of all the revolving gates in the circuit by a factor of 2, 4, and 8 to get the operators $exp[iA(2\pi/8)]$, $exp[iA(2\pi/4)]$ and $exp[iA(2\pi/2)]$. They show a 4th-order linear equation as shown in **Figure 2.** General HHL algorithm quantum circuit below. The quantum circuit of the HHL algorithm solved by the group verifies the feasibility of this circuit model to solve the linear equation system.

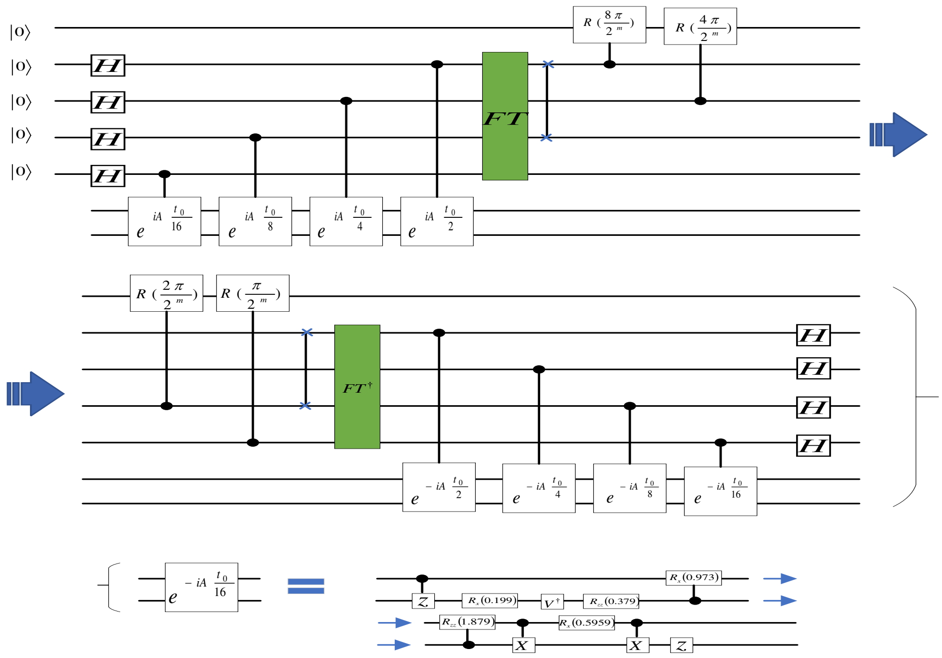


Figure 2. General HHL algorithm quantum circuit

Hybrid HHL quantum circuit implementation: Yonghae Lee et al. proposed a hybrid HHL quantum circuit implementation algorithm [6]. The implementation of the algorithm is based on the following two characteristics given in the paper:

Lemma 1. The fidelity of HHL algorithm results can reach 1 only when all eigenvalues of the matrix can perfectly *n-represented*.

For example, for a 2th-order matrix, at that time $\lambda_{n=2} = 1$, 2 or 3, its eigenvalues are *2-represented*. Using 2-bit quantum registers, its fidelity can reach 1, but if its eigenvalues exceed 3 or are decimals, it is not perfectly *2-represented*, so the fidelity cannot reach 1 when using 2-bit quantum registers.

Lemma 2. If the eigenvalues of the matrix are not perfectly *n-represented*, then the fidelity of the algorithm's results will be positively related to the number of extra quantum registers used.

This is because for matrices with non-perfect *n-represented* eigenvalues, additional qubit registers are needed to represent the eigenvalues of the matrix and to control the phase rotation. In other words, additional quantum registers are required to improve the fidelity of the results. For example, the eigenvalues of a 2th-order matrix $\lambda_{n=2} = 4$ are not perfectly represented by *2- represented*. In this case, in order to make the fidelity close to 1, it is necessary to use 3 qubits to store the eigenvalue information of the matrix.

From Lemma 1 and Lemma 2 we note that once the eigenvalues of the matrix are not perfectly *n-represented*, the number of qubits in quantum registers that need to be used to achieve high fidelity increases dramatically, even if the matrix's eigenvalues are not perfectly *n-represented*. When all eigenvalues can be perfectly *n-represented*, n -scale quantum registers are also required to ensure high fidelity. Therefore, in the implementation process of the HHL algorithm, in order to reduce the circuit complexity, it is necessary to choose a matrix with perfect *n-represented* as much as possible. At the same time, the literature **Error! Reference source not found.**pointed out the following conclusion:

Conclusion: In the execution of the HHL algorithm, for a perfect *n- represented* matrix, if it has *k-fixed* eigenvalues, it can be implemented with a smaller depth quantum circuit, and compared with the original HHL algorithm, the circuit complexity is reduced, and the smaller depth quantum circuit of HHL algorithm has higher fidelity.

Reference [6]of the AQE stage of the HHL algorithm, and gives the quantum circuit of the optimized 2th-order matrix as shown in **Figure 3.** The quantum circuits before and after the optimization of the AQE stage in the hybrid HHL algorithm. . In the hybrid quantum circuit implementation algorithm, the feedforward combined with the information obtained by classical calculation after quantum phase estimation effectively reduces the number of quantum gates of the original HHL algorithm.

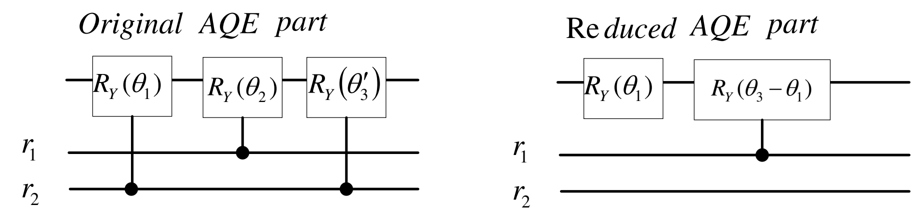


Figure 3. The quantum circuits before and after the optimization of the AQE stage in the hybrid HHL algorithm.

However, the implementation details of the optimization of the AQE stage of the HHL algorithm are not described in this algorithm, and the quantum circuits in the QPE and inverse QPE stages are not optimized.

This paper will take the general HHL algorithm quantum circuit implementation proposed by Yudong Cao et al. as the framework, and use the design idea of the hybrid HHL quantum algorithm proposed by Yonghae Lee et al. to optimize the implementation circuit of the HHL algorithm as a whole.

Result

Preparation before optimization

QPE is repeatedly performed to obtain the information of the eigenvalues: in the quantum phase estimation part, the binary representation of the eigenvalues has been stored in $Reg.B$ each qubit of the quantum register after the phase estimation. If the $Reg.B$ quantum register is measured on the Z-axis, then the $Reg.B$ quantum register can be collapsed to an eigenvalue of the matrix, and the measurement process can be repeated.

Secondly, the prior information is obtained by combining probability statistics with classical computers: whether the information of the matrix eigenvalues obtained in the QPE stage is statistically observed to be *k-fixed.* If the *k-fixed* characteristic is not observed in the statistical eigenvalue information, more qubit registers need to be used to store the matrix eigenvalues in order to discover the *k-fixed* characteristic.

Simplification of quantum circuits

In this subsection, according to the *k-fixed* property of the matrix, we give a specific implementation method to reduce the number of quantum gates required to realize the HHL quantum circuit, and compare it with the general HHL algorithm quantum circuit.

1) **Quantum Phase Estimation Stage**

After the qubits of the quantum register $Re\,g\,.C$ pass through the controll U-gate, the information of the eigenvalues of the matrix is stored in $|1\rangle's$ probability amplitude.

$$|\varphi\rangle = \frac{1}{\sqrt{2^n}}\underbrace{\left(|0\rangle + e^{2\pi i 0.\theta_{n-1}}|1\rangle\right)}_{|x_0\rangle}\underbrace{\left(|0\rangle + e^{2\pi i 0.\theta_{n-2}\theta_{n-1}}|1\rangle\right)}_{|x_1\rangle}$$

$$\ldots\underbrace{\left(|0\rangle + e^{2\pi i 0.\theta_0\ldots\theta_{n-1}}|1\rangle\right)}_{|x_{n-1}\rangle} \quad (11)$$

$0.\theta_{n-1}$, $0.\theta_{n-2}\theta_{n-1}$, … $0.\theta_0\ldots\theta_{n-1}$ is the binary expansion of $\theta_0, \theta_1, \ldots \theta_j$.

For qubits $|x_n\rangle$, n=1,2,… according to the following formula:

$$|x_n\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{\theta_0} e^{\frac{1}{2}\pi i\theta_1}\ldots e^{\frac{1}{2^{n-1}}\pi i\theta_{n-1}}|1\rangle\right) \quad (12)$$

To extract θ_0 into the qbuit, it is necessary to apply the control U-gate on $|x_n\rangle$ with a phase rotation angle of $e^{\frac{1}{2^j}\pi i\theta_j}, j = 1\ldots n-1$ before applying the H-gate to the qubit $|x_n\rangle$.

When it λ_j is *k-fixed*, as shown in $xn = \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{\theta_0} e^{\frac{1}{2}\pi i\theta_1}\ldots e^{\frac{1}{2^{n-1}}\pi i\theta_{n-1}}|1\rangle\right)$ (12), the *k-th* qubit $|x_k\rangle$ does not need to extract the eigenvalue information to the probability amplitude first like other qubits, and then store it in the quantum state. For example, when the eigenvalue of a perfect *4-represented* matrix is λ_j *2-fixed*, one can get

$$|\varphi\rangle_{\theta_1=0} = \frac{\left(|0\rangle + e^{2\pi i\left(\frac{1}{2}\theta_3\right)}|1\rangle\right)\left(|0\rangle + e^{2\pi i\left(\frac{1}{2}\theta_2+\frac{1}{4}\theta_3\right)}|1\rangle\right)}{4}$$

$$\cdot\frac{\left(|0\rangle + e^{2\pi i\left(\frac{1}{4}\theta_2+\frac{1}{8}\theta_3\right)}|1\rangle\right)\left(|0\rangle + e^{2\pi i\left(\frac{1}{2}\theta_0+\frac{1}{8}\theta_2+\frac{1}{16}\theta_3\right)}|1\rangle\right)}{4}|\mu\rangle$$

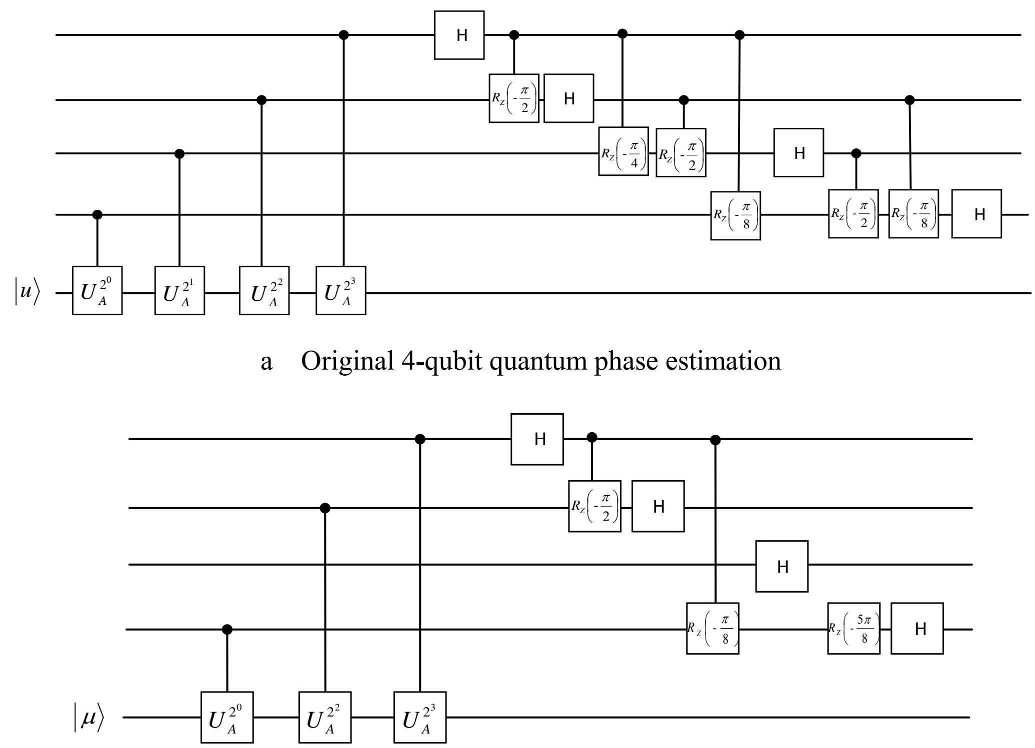
$$\rightarrow |\theta_0\theta_2\theta_3\rangle|0\rangle|\mu\rangle. \quad (13)$$

$$|\varphi\rangle_{\theta_1=1} = \frac{\left(|0\rangle + e^{2\pi i\left(\frac{1}{2}\theta_3\right)}|1\rangle\right)\left(|0\rangle + e^{2\pi i\left(\frac{1}{2}\theta_2+\frac{1}{4}\theta_3\right)}|1\rangle\right)}{4}$$

$$\cdot\frac{(|0\rangle + e^{2\pi i(\frac{1}{4}\theta_2+\frac{1}{8}\theta_3)}|1\rangle)(|0\rangle + e^{2\pi i(\frac{1}{2}\theta_0+\frac{1}{8}\theta_2+\frac{1}{16}\theta_3)}|1\rangle)}{4}|\mu\rangle$$

$$\rightarrow |\theta_0\theta_2\theta_3\rangle|1\rangle|\mu\rangle. \quad (14)$$

According to the above formula, the comparison diagram shown in **Figure 4.** Comparison of quantum phase estimation with and without *k -fixed* propertiescan be obtained.



a Original 4-qubit quantum phase estimation

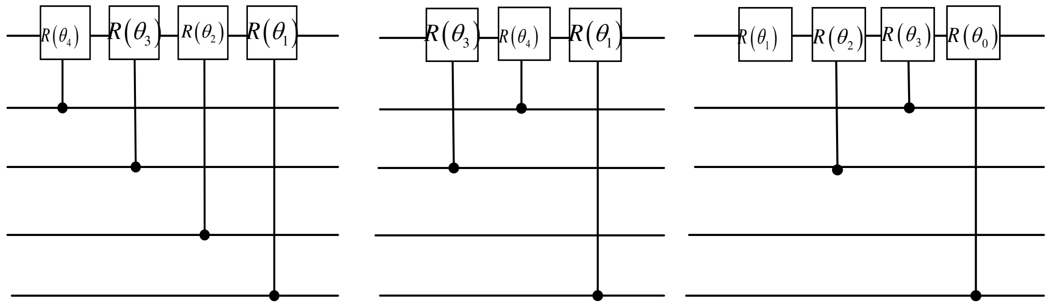
b Quantum Phase Estimation with *2-fixed* Properties

Figure 4. Comparison of quantum phase estimation with and without *k -fixed* properties

Thus, if the eigenvalues of a perfect *n-represented* matrix satisfy *k-fixed*. For a qubit representing λ_k in quantum register $Re\,g\,.C$, the applied quantum gate can be roughly reduced by a factor of $\frac{n-1}{n}$.

2) Auxiliary Quantum Encoding Phase

After phase estimation, quantum register $Re\,g\,.C$ stores a series of binary superposition states of eigenvalues. The control rotation part is to control the auxiliary qubit according to the superposition state in the quantum register at this time, as shown in Figure 5 below. **Figure 5.** AQE part of the original circuit and the optimized circuitis the quantum circuit under the general HHL algorithm quantum circuit implementation. **Figure 5.** AQE part of the original circuit and the optimized circuit is *2-fixed* and $\bar{m}_1 = 0$ time-dependent the optimized quantum circuit implementation. **Figure 5.** AQE part of the original circuit and the optimized circuit is the optimized quantum circuit implementation of 2-fixed and $\bar{m}_1 = 1$.



（a） HHL Quantum of 4th-Order Matrices quantum circuit

(b) *2-fixed* and $\bar{m}_1 = 0$ optimized quantum circuit

(c) *2-fixed* and $\bar{m}_1 = 0$ optimized quantum circuit

Figure 5. AQE part of the original circuit and the optimized circuit

After the AQE section, the system status

$$|0\rangle_A|\theta_0\theta_2\theta_3\rangle|1\rangle|\mu\rangle \rightarrow (\left(\sqrt{1-\left(\frac{c}{\lambda_{\theta_1=1}}\right)^2}\right)|0\rangle_A + \frac{c}{\lambda_{\theta_1=1}}|1\rangle_A)|\theta_0\theta_2\theta_3\rangle|1\rangle|\mu\rangle. \quad (15)$$

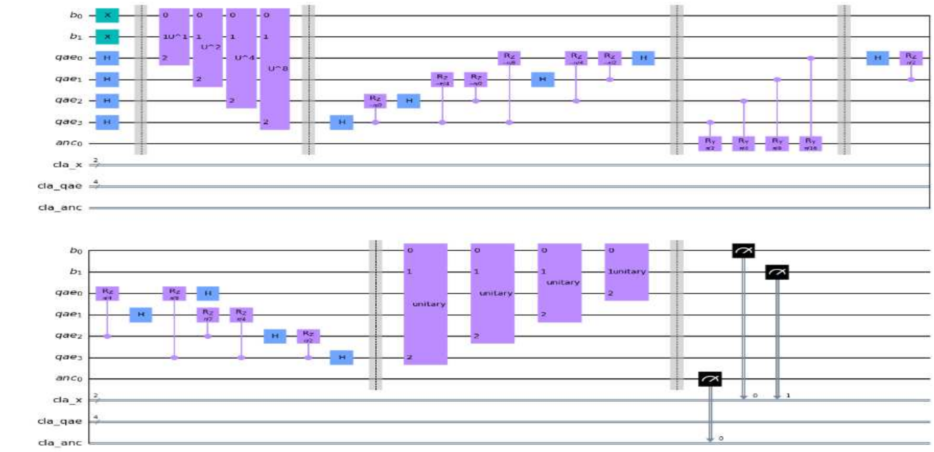
$$|0\rangle_A|\theta_0\theta_2\theta_3\rangle|0\rangle|\mu\rangle \rightarrow (\left(\sqrt{1-\left(\frac{c}{\lambda_{\theta_1=0}}\right)^2}\right)|0\rangle_A + \frac{c}{\lambda_{\theta_1=0}}|1\rangle_A)|\theta_0\theta_2\theta_3\rangle|0\rangle|\mu\rangle. \quad (16)$$

3) **Inverse quantum phase estimation**

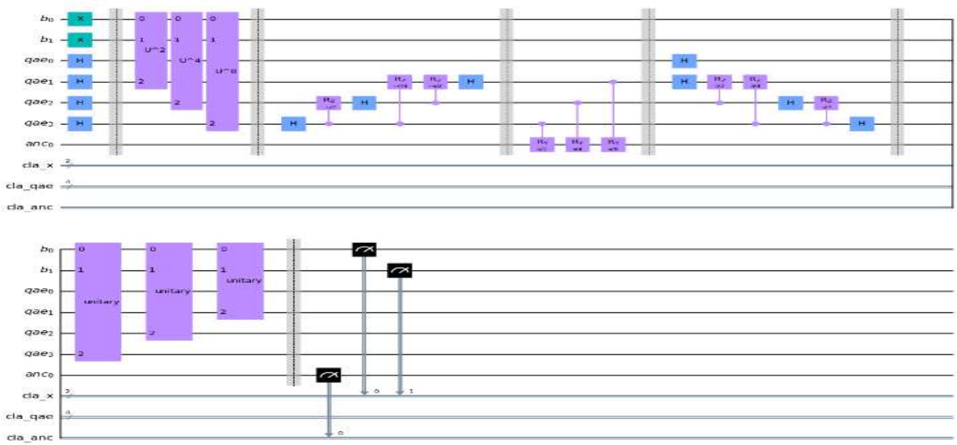
After the AQE part, the inverse quantum phase estimation and quantum phase estimation have the same simplified circuit implementation, and will not be repeated here. The applied quantum gates can likewise be roughly reduced to the original $\frac{n-1}{n}$.

Quantum circuit implementation example: IBM Q provides a Qiskit library based on the Python programming environment that can be used for remote access or emulation with classics. In this section, the Qiskit library is used to simulate the randomly selected 4th-order matrix $A=\begin{bmatrix} 11 & 5 & -1 & -1 \\ 5 & 11 & 1 & 1 \\ -1 & 1 & 11 & -5 \\ -1 & 1 & -5 & 11 \end{bmatrix}$ with the general HHL algorithm quantum circuit implementation and the global optimized HHL algorithm. Here the eigenvalues of matrix A can be accurately stored using four qubits.

The quantum circuit before and after optimization is shown in **Figure 6.** Comparison of the quantum circuit of the HHL algorithm before and after optimizationbelow.



a　General HHL algorithm implementation architecture algorithm quantum circuit diagram



b　Global optimized HHL algorithm quantum circuit diagram

Figure 6.　Comparison of the quantum circuit of the HHL algorithm before and after optimization

Before and after optimization: It can be seen from **Figure 6.** Comparison of the quantum circuit of the HHL algorithm before and after optimization that the quantum circuit of the global optimized HHL algorithm uses less quantum gates than the circuit implementation of the unoptimized HHL algorithm. On the other hand, the quantum resources and fidelity used to obtain the experimental solution using the general HHL algorithm quantum circuit implementation and the experimental solution obtained through the global optimization HHL algorithm are compared as shown in Table 1 below.

Table 1.　Comparison of Experimental and Theoretical Solutions of 4th-Order Linear Equations

Algorithm	Solution	Fidelity	Depth	Width	total quantum gate
Theoretical solution	$\begin{pmatrix}\sqrt{0.0455}\\ \sqrt{0.0455}\\ \sqrt{0.1818}\\ \sqrt{0.7272}\end{pmatrix}$	1	– –	––	––

General framework	$\begin{pmatrix}\sqrt{0.0412}\\ \sqrt{0.0450}\\ \sqrt{0.2450}\\ \sqrt{0.6687}\end{pmatrix}$	0. 993	28	14	39
Global optimization	$\begin{pmatrix}\sqrt{0.0371}\\ \sqrt{0.0358}\\ \sqrt{0.1687}\\ \sqrt{0.7583}\end{pmatrix}$	0. 998	21	14	28

Note: When using quantum circuits to implement the HHL algorithm to solve the linear equation system, the fidelity of the experimental solution cannot reach 1 due to current technical limitations.

Experimental results show that when using the global optimized HHL quantum circuit to solve linear equations, the fidelity of the experimental solution is higher than that of the original HHL general purpose quantum circuit, and it uses less quantum resources.

For hybrid HHL algorithm proposed by Yonghae Lee et al. did not give the specific implementation circuit of the entire algorithm, and when comparing the quantum resources used, they only gave the number of CNOT gates implemented by the quantum circuit of the HHL algorithm for solving 2th-order linear equations. For comparison, the method of implementation and the number of all quantum gates used are not given. For this reason, no comparison is made here.

Discussion

In general, for higher-order linear equation systems, if the prior condition is satisfied, that is, the eigenvalues of the matrix of the linear equation system have *k-fixed* characteristics, the quantum resources consumption can be reduced without reducing the fidelity of the experimental results. For an n-dimensional linear equation system, in the case of satisfying *k-fixed*, the quantum gate applied to the qubit representing λ_k in the quantum register $Re\,g\,.\,C$ can be roughly reduced to the original $\frac{n-1}{n}$, and the reduced circuit depth of the quantum circuit is about 2n.

Methods

Global optimized HHL algorithm is the same as the general HHL algorithm quantum circuit implementation in the initialization part, the difference is that the global optimized HHL algorithm uses the general HHL algorithm to achieve the QPE part, and further uses the measured $Re\,g\,.\,B$quantum register. The information of some matrix eigenvalues is used to assist the construction of the following circuit.

1) General HHL Algorithm Quantum Circuit

Initialize:

$Anc : q\,[0] \leftarrow |0\rangle$

$Re\,g\,.C:q\,[0]q[1]q[2]q[3] \leftarrow |0000\rangle$

$Re\,g\,.B:q\,[0]q[1] \leftarrow \left(U_3(\theta,0,0)\right) \otimes U_3(\theta,0,0)|00\rangle$

Different $|b\rangle$ can be achieved by changing the θ value.

QPEA:

$for\ \ i=0\ \ to\ \ 3\ \ \ do$

$$Re\,g\,.C:q\,[i] \leftarrow U_A^{2^j} \otimes (Re\,g\,.B:q\,[0]q[1], Re\,g\,.C:q\,[i])$$

end

$Re\,g\,.C:q\,[0] \leftarrow H \otimes \left(c-U_3\left(0,0,-\frac{7}{8}\pi\right)(Re\,g\,.C:q\,[3]q[2]q[1], Re\,g\,.C:q\,[0])\right)$

$Re\,g\,.C:q\,[1] \leftarrow H \otimes \left(c-U_3\left(0,0,-\frac{3}{4}\pi\right)(Re\,g\,.C:q\,[3]q[2], Re\,g\,.C:q\,[1])\right)$

$Re\,g\,.C:q\,[2] \leftarrow H \otimes \left(c-U_3\left(0,0,-\frac{1}{2}\pi\right)(Re\,g\,.C:q\,[3], Re\,g\,.C:q\,[2])\right)$

$Re\,g\,.C$:q[3] $\leftarrow H \otimes Re\,g\,.C:q\,[3]$

$c-U_3\left(0,0,-\frac{7}{8}\pi\right)$ is composed of three qubits that control different rotation angles for $Re\,g\,.C:q\,[3]q[2]q[1]$ respectively. $c-U_3(0,0,-\frac{3}{4}\pi)$ is composed of two qubits that control different rotation angles for the control bits $Re\,g\,.C:q\,[3]q[2]$ respectively.

AQE:

$anc\_q\,[0] \leftarrow c-U_3(\frac{1}{2}\pi,0,0)(Re\,g\,.C:q\,[3], anc\_q\,[0])$

$anc\_q\,[0] \leftarrow c-U_3(\frac{1}{4}\pi,0,0)(Re\,g\,.C:q\,[2], anc\_q\,[0])$

$anc\_q\,[0] \leftarrow c-U_3(\frac{1}{8}\pi,0,0)(Re\,g\,.C:q\,[1], anc\_q\,[0])$

$anc\_q\,[0] \leftarrow c-U_3(\frac{1}{16}\pi,0,0)(Re\,g\,.C:q\,[0], anc\_q\,[0])$

***Inverse_AQE*:**

$Reg.C:q[0]q[1]q[2]q[3] \leftarrow q\_f\,t(Reg.C:q[0]q[1]q[2]q[3])$

$for\ \ i=0\ \ to\ \ 3\ \ \ do$

$$Reg.C:q[i] \leftarrow inverse\_U_A^{2^i} \otimes (Reg.B:q[0]q[1], Reg.C:q[i])$$

end

Measure:

$Do\ \ measure\ \ anc\_q\,[0]$

$if\ \ anc_q\,[0] = |1\rangle$then

$return\ \ Reg.B:q[0]q[1]$

end

2) Globally optimized HHL algorithm

Initialization:

$Anc:q\,[0] \leftarrow |0\rangle$

$Re\,g\,.C:q\,[0]q[1]q[2]q[3] \leftarrow |0000\rangle$

$Re\,g\,.B:q\,[0]q[1] \leftarrow \left(U_3(\theta,0,0)\right) \otimes U_3(\theta,0,0)|00\rangle$

QPEA:

$for\ \ i=1\ \ to\ \ 3\ \ \ do$

$Re\,g\,.C:q\,[i] \leftarrow U_A^{2^j} \otimes (Re\,g\,.B:q\,[0]q[1], Re\,g\,.C:q\,[i])$

end

$Re\,g\,.C:q\,[1] \leftarrow H \otimes \left(c - U_3\left(0,0,-\frac{3}{4}\pi\right)(Re\,g\,.C:q\,[3]q[2], Re\,g\,.C:q\,[1])\right)$

$Re\,g\,.C:q\,[2] \leftarrow H \otimes \left(c - U_3\left(0,0,-\frac{1}{2}\pi\right)(Re\,g\,.C:q\,[3], Re\,g\,.C:q\,[2])\right)$

$Re\,g\,.C:$ q[3] $\leftarrow H \otimes Re\,g\,.C:q\,[3]$

AQE:

$anc\_q\,[0] \leftarrow c - U_3(\frac{1}{2}\pi,0,0)(Re\,g\,.C:q\,[3], anc\_q\,[0])$

$anc\_q\,[0] \leftarrow c - U_3(\frac{1}{4}\pi,0,0)(Re\,g\,.C:q\,[2], anc\_q\,[0])$

$anc\_q\,[0] \leftarrow c - U_3(\frac{1}{8}\pi,0,0)(Re\,g\,.C:q\,[1], anc\_q\,[0])$

$\boldsymbol{Inverse\_AQE}$:

$Reg.C: q[0]q[1]q[2]q[3] \leftarrow q\_f\,t(Reg.C: q[0]q[1]q[2]q[3])$

$for\ \ i=1\ \ to\ \ 3\ \ \ do$

$Reg.C: q[i] \leftarrow inverse\_U_A^{2^i} \otimes (Reg.B: q[0]q[1], Reg.C: q[i])$

end

Measure:

$Do\ \ measure\ \ anc\_q\,[0]$

$if\ \ anc_q\,[0] = |1\rangle$then

$return\ \ Reg.B: q[0]q[1]$

end

Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

References:

[1] Shor P W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer[J]. Siam Review, 1999, 41(2):303-332.

[2] Grover L K. Quantum Mechanics Helps in Searching for a Needle in a Haystack[J]. Physical Review Letters, 1997, 79(2).

[3] Harrow A W, Hassidim A, Lloyd S. Quantum Algorithm for Linear Systems of Equations[J]. Physical Review Letters, 2009, 103(15):150502-150502.

[4] Maslov D, Dueck GW, Miller DM, et al. Quantum circuit simplification and level compaction[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2008, 27(3): 436-444.

[5] Cory D G, Laflamme R, Knill E, et al. ChemInform Abstract: NMR Based Quantum Information Processing: Achievements and Prospects[J]. ChemInform, 2001, 32(7).
[6] Lee Y, Joo J, Lee S. Hybrid quantum linear equation algorithm and its experimental test on IBM Quantum Experience[J]. Scientific Reports, 2019, 9(1).
[7] Langenberg B, Pham H, Steinwandt R. Reducing the cost of implementing the advanced encryption standard as a quantum circuit[J]. IEEE Transactions on Quantum Engineering, 2020, 1: 1-12.
[8] Zou J, Wei Z, Sun S, et al. Quantum circuit implementations of AES with fewer qubits[C]//International Conference on the Theory and Application of Cryptology and Information Security. Springer, Cham, 2020: 697-726.
[9] Jaques S, Naehrig M, Roetteler M, et al. Implementing Grover oracles for quantum key search on AES and LowMC [C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Cham, 2020: 280-310.
[10] Monz T, Nigg D, Martinez EA, et al. Realization of a scalable Shor algorithm[J]. Science, 2016, 351(6277): 1068-1070.
[11] Diao Z, Zubairy MS, Chen G. A quantum circuit design for Grover's algorithm[J]. Zeitschrift fur Naturforschung A, 2002, 57(8): 701-708.
[12] Grassl M, Langenberg B, Roetteler M, et al. Applying Grover's algorithm to AES: quantum resource estimates[C]//Post-Quantum Cryptography. Springer, Cham, 2016: 29-43.
[13] Ambainis A. Variable time amplitude amplification and a faster quantum algorithm for\n, solving systems of linear equations[J]. Computer Science, 2010.
[14] Cao Y, Daskin A, Frankel S, et al. Quantum circuit design for solving linear systems of equations[J]. Molecular Physics, 2012, 110(15-16):1675-1680.
[15] Gado M, Younes A. Optimization of Reversible Circuits Using Toffoli Decompositions with Negative Controls[J]. Symmetry, 2021, 13(6):1025.
[16] Storme L, Vos AD, Jacobs G. Group Theoretical Aspects of Reversible Logic Gates[J]. Journal of Universal Computer Science, 1999, 5(5):307-321.
[17] He X, Guan Z, Ding F. The Mapping and Optimization Method of Quantum Circuits for Clifford + T Gate[J]. Applied Mathematics and Applied Physics (English), 2019, 7(11):15.
[18] Mamun M, D Menville. Quantum Cost Optimization for Reversible Sequential Circuit[J]. International Journal of Advanced Computer Science and Applications, 2014, 4(12):15-21.
[19] Daskin A, Kais S. Decomposition of unitary matrices for finding quantum circuits: Application to molecular Hamiltonians[J]. Journal of Chemical Physics, 2011, 134(14):2594-125.
[20] Daskin A, Kais S. Group Leaders Optimization Algorithm[J]. Molecular Physics, 2011, 109(5):761-772.
[21] https://qiskit.org/textbook/ch-applications/hhl_tutorial.html

Additional Information

Competing Interests: The authors declare no competing interests.

Acknowledgements

The authors would like to thank for the support from the National Science Foundation of China (No.61941105). We acknowledge use of the qiskit for this work. Te views expressed are those of the authors and do not refect the ofcial policy or position of IBM or the qiskit team.

Author Contributions

M.Z., L.D. and Y.Z. conceived the experimental test on IBM Quantum Experience, and M.Z. conducted the experimental test. M.Z. and L.D. wrote the manuscript. Y.Z. and N.C. supervised the work and revised the manuscript. All authors discussed the results, and reviewed the manuscript.