

# Experimental Challenges of Implementing Quantum Phase Estimation Algorithms on IBM Quantum Computer

Hamed Mohammadbagherpoor<sup>1</sup>, Young-Hyun Oh<sup>2</sup>, Anand Singh<sup>2</sup>, Xianqing Yu<sup>2</sup>, Andy J. Rindos<sup>2</sup>

<sup>1</sup>Department of Electrical and Computer Engineering,  
North Carolina State University, Raleigh, NC 27606, USA

<sup>2</sup>Hybrid Cloud, IBM, Durham, NC, USA

**Abstract**—Many researchers have been heavily investigated on quantum phase estimation (QPE) algorithms to find the unknown phase, since QPE is the core building block of the most quantum algorithms such as the Shor’s factoring algorithm, quantum sampling algorithms, and finding the eigenvalues of unitary matrices. Kitaev’s algorithm and QPE algorithms using inverse Quantum Fourier transform were proposed and widely used by researchers as a key component for their quantum algorithms. In this paper, we explore the experimental challenges of QPE algorithms on Noisy Intermediate-Scale Quantum (NISQ) computers by implementing various QPE algorithms on the state-of-the-art IBM quantum computer. Our experimental results demonstrate that the accuracy of finding the phase using these algorithms are severely constrained by NISQ’s physical characteristics such as coherence time and error rates. To mitigate such physical limitations, we propose modified solutions of these algorithms by reducing the number of control gates and phase shift operations. Our experimental results showed that our solutions can significantly increase the accuracy of the finding phase in near-term quantum computers.

## I. INTRODUCTION

Noisy Intermediate-Scale Quantum (NISQ) [1] computers provide a framework which can solve particular problems exponentially faster than classical computers by taking advantages of the computational power coming from the use of superposition and entanglement of the qubits [2]. Classical computers works based on a Load-Run-Read cycle in which the input data is loaded into the system, a program runs and then the output of the program is read. However, in quantum computers there is Prepare-Evolve-Measure cycle wherein the quantum states are prepared as the input, manipulate the input states in quantum computers using the operators and then the results are measured. [3]. Recently, remarkable progress on quantum computers has been achieved by different industries that enable researchers to implement particular classical algorithms using publicly available near-term quantum computers such as IBM QX [3] and Rigetti QPU [4].

Quantum phase estimation (QPE) is the critical building block for various quantum algorithms such as Shor’s algorithm for factoring the prime numbers, quantum chemistry to model the molecules and, Grover’s algorithm to search [6]–[12]. However, implementing QPE algorithms on near-term quantum computers are severely constrained by low reliability and high variability of quantum computers’ physical characteristics. For example, Shor’s algorithm for factoring 15 (i.e.,  $3 \times 5$ ) on a nuclear magnetic resonance (NMR) computer is presented in [13] and the number 21 is factored by imple-

menting qubit recycling in a photonic circuit [13]. The largest number factored by actual quantum computer is the number 143 which was implemented on a dipolar coupling NMR System by applying adiabatic quantum computation [15]. Although near-term quantum computers are currently limited to a small number of qubits, such experimental approaches will be considerably valuable when we can take full advantage of quantum supremacy in near future.

The main objective of quantum phase estimation is to determine the eigenvalues of a unitary matrix with an unchanged eigenvector. Two main approaches are used to implement quantum phase estimation; The first approach is to extract the phase information by applying the classical post processing computation after utilizing quantum gate operations as known as Kitaev’s algorithm [16], [17]. Since Kitaev’s algorithm requires some classical post processing after performing Hadamard tests, it is necessary to run a minimal number of trials of Hadamard tests to obtain the phase  $k_{th}$  bit position with constant success probability. The other approach is to find the phase information using Quantum Fourier transform in which the phase is estimated by applying quantum inverse Fourier Transform [18], [19], [22]. An experimental phase estimation based on quantum Fourier transform was implemented on a three-bit nuclear magnetic resonance (NMR) processor [20] but it only used to estimate the eigenvalues of one-bit Grover operators. An implementation of phase estimation algorithm on an ion-trapped quantum computer was proposed to find the eigenstates of the system [21]. Lloyd et al. have been shown that quantum computers can speed up of some linear algebraic based machine learning algorithms by applying quantum phase estimation technique such as principle component(PCA), support vector machine (SVM), and K-means algorithms [23], [24]. However, QFT approach requires a large number of rotation gates for precision digits to obtain more accurate phase information. Without loss of generality, this means that more rotations gates can dramatically increase readout errors from implementation results of QFT algorithms on near-term quantum computers. Thus, it is fundamental to minimize depth and controlled-rotation gates to increase the accuracy of finding the phase on quantum computers.

In this paper, we investigate on various quantum phase estimation algorithms but, based on our literature review, it is difficult to find experimental results of QPE algorithms from NISQ computers. To address the lack of experimental results, we implement these algorithms on both IBM QASM simulator

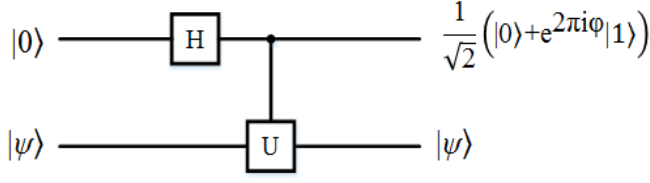


Fig. 1: Quantum circuit for transforming the states

and the state-of-the-art IBM QX machine and then analyze experimental challenges of implementing quantum algorithm on real quantum computers. The experimental results show that the accuracy of finding the correct phase decreases as the number of qubits and quantum operations increase. To mitigate the problem, we propose modified solutions of these QPE algorithms by minimizing the number of controlled-gate and phase shift operators. Our experimental results demonstrate that our solutions significantly increase the accuracy of finding correct phase.

This paper is categorized as follows. Section II describes the basic quantum operations and various phase estimation algorithms such as Kitaev's algorithm, iterative algorithm to estimate the phase, and Lloyd algorithm for phase estimation based on inverse Fourier transform (QFT). In section III, the simulation and experimental results for each method are provided and compared. Finally, the conclusion of the paper is summarized in Section IV.

## II. PHASE ESTIMATION

Quantum phase estimation of an unitary matrix can be mainly derived using two methods, (1) applying the inverse Quantum Fourier Transform (QFT) to derive the unknown phase information and (2) sequential post processing techniques to calculate the unknown phase. Phase estimation technique is used to estimate the eigenvalues  $|\lambda\rangle$  of an unitary matrix  $U$  with its known eigenvector  $|\psi_\lambda\rangle$  [2],

$$U|\psi_\lambda\rangle = \lambda|\psi_\lambda\rangle \quad (1)$$

where eigenvalues of the unitary matrix is  $\lambda = e^{2\pi i\phi_m}$ . The goal of phase estimation is to find the eigenvalues of the unitary matrix and then applying them to estimate the unknown phase of the unitary operator. Let the phase of the unitary matrix be  $\phi_k = 0.x_1x_2x_3\dots x_n$  where  $n$  is the number of qubits used for phase estimation. The estimated variable ( $\hat{\phi}$ ) can be expressed as a binary representation,

$$\hat{\phi} = \frac{x_1}{2} + \frac{x_2}{2^2} + \frac{x_3}{2^3} + \dots + \frac{x_n}{2^n} \quad (2)$$

Fig.1 shows the quantum circuits are applied to one qubit and eigenstate. The output of the circuits contains the phase described in the top of the Fig.1 but it is impossible to find the correct phase due to the superposition state on the value. In order to estimate the phase, it requires to apply different QPE techniques to provide the information about the phase of the system. In the next sections, we will explain these techniques

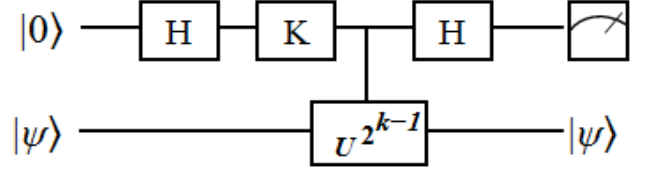


Fig. 2: Controlled U circuit

along with their simulated and experimentally implemented results.

### A. Kitaev's Algorithm

Kitaev's algorithm is the first algorithm that was introduced to estimate the phase of an unitary matrix. In this technique a set of Hadamard gates are applied to the input qubits, the outputs of Hadamard gate are performed with Controlled- $U^{2^{k-1}}$  to implement the phase shift operator. Applying controlled- $U$  operator  $k$  times transforms the control qubit to  $\frac{1}{\sqrt{2}}(|0\rangle + e^{-i2\pi\phi_k 2^{k-1}}|1\rangle)$ . At each test phase  $\phi_k = 2^{k-1}\phi$  can be calculated. By doing  $k$  times test and measuring the output of each test the set of values  $\phi, 2\phi, \dots, 2^{k-1}\phi$  can be achieved. These measurements are used to estimate the phase of the unitary matrix. Fig.2 shows the circuit to perform phase estimation. As can be seen for the circuit operation  $K$  can be used to manipulate the qubit phase and provides more information about the phase of the system. Considering  $K = I_2$ , the quantum circuit provide the following analysis,

$$\begin{aligned} |0\rangle|\psi_\lambda\rangle &\xrightarrow{H\otimes I} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|\psi_\lambda\rangle \\ &\xrightarrow{C-U_k} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes U_k|\psi_\lambda\rangle \\ &= \frac{1}{\sqrt{2}}(|0\rangle|\psi_\lambda\rangle + e^{2\pi i\phi_k}|1\rangle|\psi_\lambda\rangle) \\ &\xrightarrow{H\otimes I} \frac{1}{\sqrt{2}}\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}|\psi_\lambda\rangle + \frac{e^{2\pi i\phi_k}(|0\rangle-|1\rangle)}{\sqrt{2}}|\psi_\lambda\rangle\right) \\ &= \frac{1}{2}\left(\left(1+e^{2\pi i\phi_k}\right)|0\rangle + \left(1-e^{2\pi i\phi_k}\right)|1\rangle\right)|\psi_\lambda\rangle \end{aligned} \quad (3)$$

Based on the calculations from Eq.3, the probability of measuring  $|0\rangle$  and  $|1\rangle$  will be,

$$P(0|k) = \frac{1 + \cos(2\pi\phi_k)}{2}, \quad P(1|k) = \frac{1 - \cos(2\pi\phi_k)}{2} \quad (4)$$

$\phi_k$  can be obtained with more precise estimated digit by applying more trials. However, based on the data from Eq. 3 we cannot distinguish between  $\phi_k$  and  $-\phi_k$ . Another circuit is required to be considered to provide more information about the phase of unitary matrix and helps to distinguish between  $\phi_k$  and  $-\phi_k$ . Combination of the results from  $K = I_2$  and  $K = S$

helps to find the actual value of the phase. In a case that  $K = S$  gate is used and applied in the circuit, the analysis will be,

$$K = S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad (5)$$

Quantum circuit provides the following transformation,

$$\begin{aligned} |0\rangle |\psi_\lambda\rangle &\xrightarrow{H \otimes I} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) |\psi_\lambda\rangle \\ &\xrightarrow{S} \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) |\psi_\lambda\rangle \\ &\xrightarrow{C-U_k} \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \otimes U_k |\psi_\lambda\rangle \end{aligned} \quad (6)$$

$$= \frac{1}{\sqrt{2}}(|0\rangle |\psi_\lambda\rangle + ie^{2\pi i \phi_k} |1\rangle |\psi_\lambda\rangle)$$

$$\xrightarrow{H \otimes I} \frac{1}{\sqrt{2}} \frac{(|0\rangle + i|1\rangle)}{\sqrt{2}} |\psi_\lambda\rangle + i \frac{e^{2\pi i \phi_k} (|0\rangle - i|1\rangle)}{\sqrt{2}} |\psi_\lambda\rangle$$

$$\begin{aligned} &= \frac{1}{2} \left( (1 + ie^{2\pi i \phi_k}) |0\rangle + (1 - ie^{2\pi i \phi_k}) |1\rangle \right) |\psi_\lambda\rangle \\ &= \frac{1}{2} \left( (1 + e^{2\pi i \phi_k + \frac{\pi}{2}}) |0\rangle + (1 - ie^{2\pi i \phi_k + \frac{\pi}{2}}) |1\rangle \right) |\psi_\lambda\rangle \end{aligned} \quad (7)$$

The probabilities in this case will be,

$$P(0|k) = \frac{1 - \sin(2\pi \phi_k)}{2}, \quad P(1|k) = \frac{1 + \sin(2\pi \phi_k)}{2} \quad (8)$$

Eq.8 provides more information about the phase that helps to recover the phase of the unitary matrix. In each test the probabilities of being zero or one in  $t$  trials are measured. By using the result from Eq.4 and Eq.8 the estimation of  $\cos(2\pi \phi_k)$  and  $\sin(2\pi \phi_k)$  and as a result phase ( $\hat{\phi}$ ) can be calculated as,

$$\hat{\phi}_k = \frac{1}{2\pi} \tan^{-1} \left( \frac{C_k}{S_k} \right) \quad (9)$$

where  $C_k$  and  $S_k$  are the estimation of  $\cos(2\pi \phi_k)$  and  $\sin(2\pi \phi_k)$  respectively. In Kitaev's algorithm post processing calculation is required to estimate the value of the phase. Estimating of the phase within  $m$  bits of accuracy requires to increase the number of trials.  $O\left(\frac{\log(1-\delta)}{\epsilon}\right)$  samples are required to estimate within  $\epsilon$  with probability of  $1 - \delta$ .

---



---

#### Iterative Phase estimation

---



---

- 1: for  $i = 1$  to  $m$  do
  - 2: Choose Controlled-U gate.
  - 3: Perform basic Kitaev's circuit and measure the output
  - 4: Calculate  $i^{\text{th}}$  bit ( $\phi_k$ ) of the phase by applying the information from previous iteration result ( $\phi_{k-1}$ )
  - 5: Update Controlled-U gate.
  - 6: end for
  - 7: Return the estimation of the phase.
- 

TABLE I: Iterative Quantum phase estimation

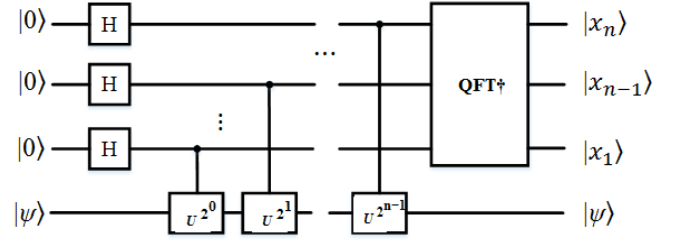


Fig. 3: Quantum Phase estimation based on inverse Quantum Fourier Transform (QFT)

#### B. Iterative Quantum Phase estimation

In the experimental implementation, increasing the number of gate to estimate the phase with higher accuracy will increase the convergence error and approach to a wrong answer. In this section an iterative technique was proposed to implement the Kitaev's algorithm with higher accuracy and applying two qubits. This sequential approach helps to find the unknown phase of the system with  $m$  bits of accuracy. Table I shows the the proposed algorithm. As it can be seen, one Hadamard gate was used to perform the superposition and then U-controlled gate is applied to the output of the Hadamard gate then, the output of this stage transform to the measurement state by applying another Hadamard gate. In the next iteration the order of Controlled-U gate is updated and the result from the previous measurement was used and applied to the circuit to estimate the new bit. This technique was repeated  $m$  times to estimate the phase with  $m$  bits of accuracy. In this method in each iteration the information from the previous iterations were used to estimated the next bit of the phase

#### C. Phase estimation based on inverse QFT

One of the common method that is used to implement QPE is based on inverse QFT. The general view of this method has been shown in Fig.3. In this method two stages are required for phase estimation. First stage starts with  $n$ -qubits initialized at  $|0\rangle$ , prepares the state  $|\psi\rangle$  and, the second stage uses inverse quantum Fourier transform operation to estimate the binary digits of the phase. The mathematical analysis of the first stage is as,

$$\begin{aligned} &\frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i 2^{n-1} \phi} |1\rangle) \frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i 2^{n-2} \phi} |1\rangle) \dots \\ &\frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i \phi} |1\rangle) = \frac{1}{2^{n/2}} \sum_{k=0}^{2^n-1} e^{2\pi i \frac{\phi k}{2^n}} |k\rangle \end{aligned} \quad (10)$$

considering  $\phi = x/2^n$  where  $x = \sum_{i=0}^{n-1} 2^i x_i$  we have,

$$\begin{aligned} &\frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i 0.x_n} |1\rangle) \frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i 0.x_{n-1}x_n \phi} |1\rangle) \dots \\ &\frac{1}{\sqrt{2}} (|0\rangle + e^{2\pi i 0.x_1x_2 \dots x_n} |1\rangle) = \frac{1}{2^{n/2}} \sum_{k=0}^{2^n-1} e^{2\pi i \frac{\phi k}{2^n}} |k\rangle \end{aligned} \quad (11)$$

As can be seen from Fig.3 the outputs from the first stage (phase Kick-back) are the input of inverse QFT. By applying controlled- $U^{2^{n-1}}$  there will phase kick back to prepare the states. Also, the output of the first stage is exactly quantum

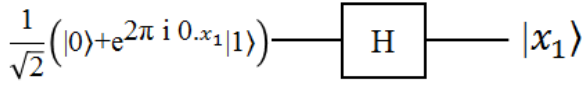


Fig. 4: One bit phase estimation Quantum circuit

Fourier transform of  $\varphi$ . By applying the inverse QFT we can recover the unknown phase. In order to analyze this method two different phase estimation circuits with different accuracies have been considered,

**Case1:** Considering  $\varphi = 0.x_1$ , the circuit shown in Fig. 4 and applying Hadamard gate to the initial state  $|0\rangle$  we have:

$$\begin{aligned}
 |0\rangle &\xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\
 &\xrightarrow{U} \frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i\varphi}|1\rangle) \\
 &\xrightarrow{H} \frac{1}{2}(1 + e^{2\pi i\varphi})|0\rangle + (1 - e^{2\pi i\varphi})|1\rangle \\
 &= \frac{1}{2}(1 + e^{2\pi i0.x_1})|0\rangle + (1 - e^{2\pi i0.x_1})|1\rangle
 \end{aligned} \tag{12}$$

Calculating the probability from equation (12) we have,

$$P(|0\rangle) = \frac{1 + \cos(2\pi 0.x_1)}{2}, \quad P(|1\rangle) = \frac{1 - \cos(2\pi 0.x_1)}{2} \tag{13}$$

Based on the result from equation (13), if  $x_1 = 0$ , then the probability of  $|0\rangle$  is 1 ( $P(|0\rangle) = 1$ ) and if  $x_1 = 1$ , then the probability of  $|1\rangle$  is 1 ( $P(|1\rangle) = 1$ ). It means that in a case that phase is considered as one bit only one Hadamard gate is required to extract  $x_1$ .

**Case2:** Considering  $\varphi = 0.x_1x_2$ , implementing circuit in Fig.5 and applying inverse QFT, the unknown phase can be derived. The second digit ( $x_2$ ) can be extracted by applying one Hadamard gate, the same as case 1. In order to extract the first digit ( $x_1$ ), a controlled-rotation gate  $R_2$  is required to remove the impact of the  $x_2$  and as a result convert it to case 1 and use one Hadamard gate to estimate  $x_1$ , as,

$$\begin{aligned}
 &\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i0.x_1x_2}|1\rangle) \xrightarrow{C-R^*2} \\
 &\quad \frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i(x_1*2^{-1} + x_2*2^{-2} - x_2*2^{-2})}|1\rangle) \\
 &= \frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i x_1 * 2^{-1}}|1\rangle) \xrightarrow{H} \\
 &\frac{1}{\sqrt{2}}(1 + e^{2\pi i0.x_1})|0\rangle + (1 - e^{2\pi i0.x_1})|1\rangle
 \end{aligned} \tag{14}$$

Calculating the probability from equation (14) we have,

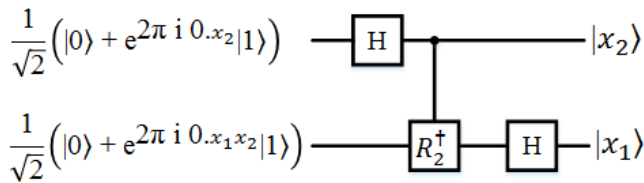


Fig. 5: 3-Qubits inverse Quantum Fourier Transform (QFT)

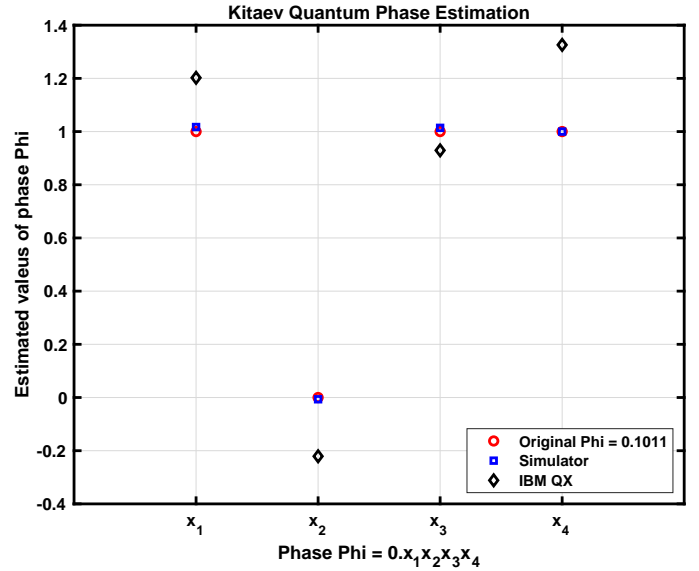


Fig. 6: Kitaev Quantum Phase Estimation Algorithm on Simulator and IBM QX

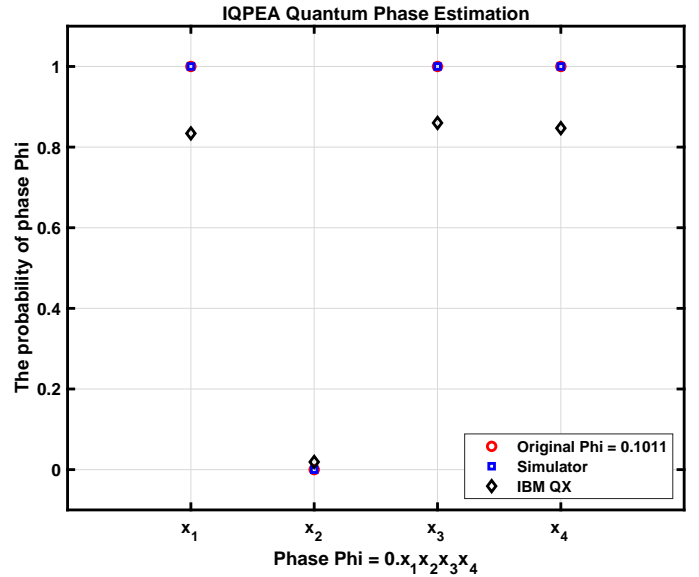


Fig. 7: Iterative Quantum Phase Estimation Algorithm on Simulator and IBM QX

$$P(|0\rangle) = \frac{1 + \cos(2\pi 0.x_1)}{2}, \quad P(|1\rangle) = \frac{1 - \cos(2\pi 0.x_1)}{2} \tag{15}$$

The rotation gate  $R_2$  is defined as Eq.(16) where  $k = 2$ .

$$R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{pmatrix} \tag{16}$$

### III. SIMULATION AND EXPERIMENT RESULTS

In this section, we show how to implement various Quantum Phase Estimation (QPE) algorithms on both IBM QASM simulator and the state-of-the-art IBM QX 4 machine (i.e., ibmqx4). Our results show that the accuracy of their experimental results is significantly reduced as the number of qubits increases. For example, single qubit in IBM Q

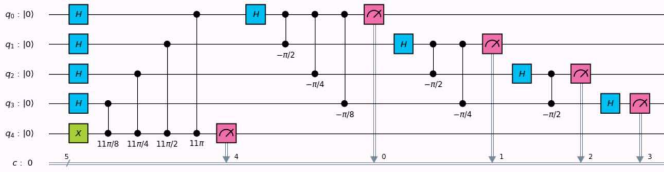


Fig. 8: Lloyd QPE Algorithm Gate with 1 ancillary qubit

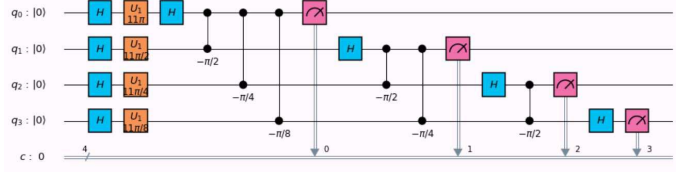


Fig. 10: Modified Lloyd QPE Algorithm Gate without 1 ancillary qubit

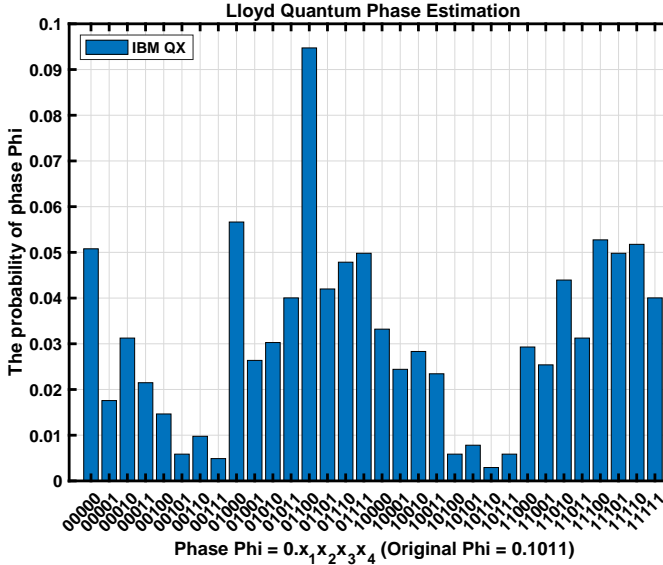


Fig. 9: Lloyd QPE Algorithm on IBM QX (ibmqx4)

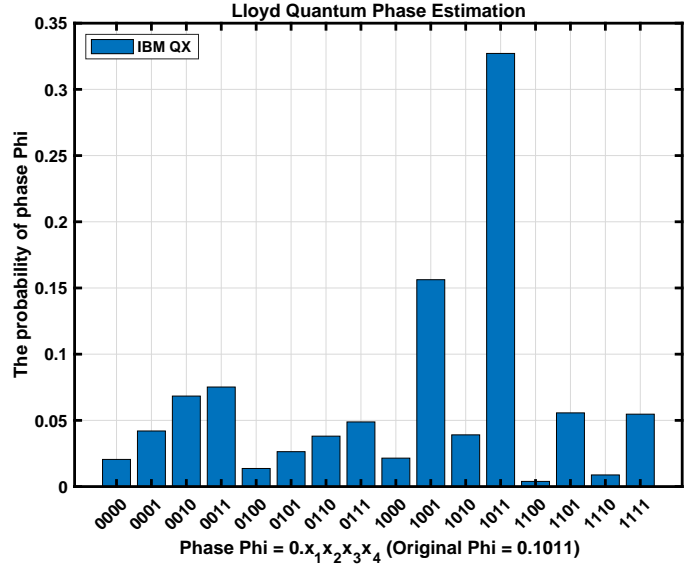


Fig. 11: Modified Lloyd QPE Algorithm on IBM QX (ibmqx4)

experience has good fidelity on most quantum operations but the fidelity will be quickly decreased as the number of control qubits increases. To mitigate the problem, we implement our modified solutions of these QPE algorithms in order to increase the accuracy of the finding phase. Our solutions not only carefully design quantum gates of QPE algorithms but also take the advantages of classical computers' capabilities such as storing intermediate results to feed the values into the next quantum operations. Our simulation results demonstrate that our solutions dramatically increase the accuracy of finding the correct phase. In our experiments, we set the phase is  $\varphi = 0.x_1x_2x_3x_4$ , where the number of phase bit positions is 4 ( $n = 4$ ). We defined  $\varphi = 1/2 + 1/8 + 1/16$  which represents  $\varphi = 0.1011$  as a binary value. For each QPE algorithm, we ran the default 1,024 shots for both simulator and IBM QX machine.

First, we implement Kitaev's algorithm to find the phase  $\varphi$  on both IBM QASM Simulator and IBM QX quantum computer. Fig.6 shows that the estimated  $\hat{\varphi}$  values of simulator results are almost the same as the original  $\varphi$  values. The estimated  $\varphi$  values from IBM experiment results are slightly different than the original  $\varphi$  due to the lack of full error correction capability of quantum computers yet. However, we can estimate the correct binary values of bit positions by converting the estimated  $\varphi$  values. Since the noise can be various among different quantum computers, it is critical to find the hardware error rates in order to increase the accuracy of the  $\varphi$  estimation in Kitaev's algorithm. The accuracy can

be increased by adjusting proper error rates for each quantum computer during the computation process from the estimated  $\varphi$  into the binary bit position.

Second, we implemented iterative quantum phase estimation algorithm (IQPEA) to find the phase  $\varphi$  on both IBM Qasm Simulator and IBM QX quantum computer. Fig.7 shows that the probability of finding  $\varphi$  value from the simulation results are exactly the same as the original  $\varphi$ . The experiment results are slightly different than the original  $\varphi$  but we can estimate the correct binary values the same way as Kitaev's algorithm.

Last, we implemented QPE algorithms using inverse Quantum Fourier Transform technique to find the phase  $\varphi$  on both IBM Qasm Simulator and IBM QX quantum computer. Fig.9 only shows the probability of finding  $\varphi$  values from IBM QX experiment results because the probability of finding  $\varphi$  value from the simulation results are exactly the same as the original  $\varphi$ . However, the highest probability of the phase  $\varphi$  from the experimental results is when the  $\varphi$  is 0.01100 instead of the correct  $\varphi = 0.1011$ . The main reasons for these incorrect results are caused by the lack of error correction capability, short longitudinal, and transversal coherence time for qubits and ancillary qubits respectively. Moreover, as described in Fig.8, the number of controlled phase rotation gates on qubits can increase the readout errors. To solve this problem and increase the accuracy of experimental results, we remove the ancillary control qubit and replace the unnecessary controlled-rotation gates with unitary rotation gates for each qubit as described in Fig.10. Our experimental results Fig.11 shows

that our solution can find the correct phase  $\varphi$  and even the probability (i.e., 0.335%) is completely distinguished from other estimated  $\varphi$  values.

#### IV. CONCLUSIONS

This paper investigates on the existing Quantum Phase Estimation (QPE) algorithms and explains how to implement these algorithms on the state-of-the-art IBM quantum computer. We discover the challenges of implementing QPE algorithms on real quantum processor and propose modified solution of these algorithms by minimizing the number of controlled-rotation gates and by utilizing the classical computer's capabilities. Our experimental results can guide researchers to consider these challenges when they implement their quantum algorithms on Noisy Intermediate-Scale Quantum (NISQ) computers before we can fully take the advantages of quantum supremacy in near future.

#### REFERENCES

- [1] John Preskill, Quantum Computing in the NISQ era and beyond, arXiv:1801.00862.
- [2] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information (Cambridge University Press, Cambridge, 2000).
- [3] Rigetti, <https://www.rigetti.com/>.
- [4] IBM Q Experience, <https://quantumexperience.ng.bluemix.net/qx>.
- [5] C. P. Williams, Explorations in Quantum Computing. Springer London, 2011.
- [6] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Science 309, 1704 (2005).
- [7] B. P. Lanyon et al., Towards quantum chemistry on a quantum computer, Nature Chemistry, vol. 2, no. 2, pp. 106111, Jan. 2010.
- [8] P. W. Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, SIAM Journal on Computing, vol. 26, no. 5, pp. 14841509, Oct. 1997.
- [9] K. Temme, T. J. Osborne, K. G. Vollbrecht, D. Poulin, and F. Verstraete, Quantum Metropolis sampling, Nature, vol. 471, no. 7336, pp. 8790, Mar. 2011.
- [10] M. Ozols, M. Roetteler, and J. Roland, Quantum rejection sampling, in Proceedings of the 3rd Innovations in Theoretical Computer Science Conference on - ITCS 12, 2012.
- [11] L. M. K. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, M. H. Sherwood, and I. L. Chuang, Experimental realization of Shors quantum factoring algorithm using nuclear magnetic resonance, Nature, vol. 414, no. 6866, pp. 883887, Dec. 2001.
- [12] A. Politi, J. C. F. Matthews, and J. L. O'Brien, Shors Quantum Factoring Algorithm on a Photonic Chip, Science, vol. 325, no. 5945, pp. 12211221, Sep. 2009.
- [13] L. M. K. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, M. H. Sherwood, and I. L. Chuang, Experimental realization of Shors quantum factoring algorithm using nuclear magnetic resonance, Nature, vol. 414, no. 6866, pp. 883887, Dec. 2001.
- [14] X. Peng et al., Quantum Adiabatic Algorithm for Factorization and Its Experimental Implementation, Physical Review Letters, vol. 101, no. 22, Nov. 2008.
- [15] N. Xu, J. Zhu, D. Lu, X. Zhou, X. Peng, and J. Du, Quantum Factorization of 143 on a Dipolar-Coupling Nuclear Magnetic Resonance System, Physical Review Letters, vol. 108, no. 13, Mar. 2012.
- [16] A. Y. Kitaev, Quantum computations: algorithms and error correction, Russian Mathematical Surveys, vol. 52, no. 6, pp. 11911249, Dec. 1997.
- [17] A. Y. Kitaev, A. Shen, and M. Vyalii, Classical and Quantum Computation (American Mathematical Society, Providence, Rhode Island, 2002).
- [18] D. S. Abrams and S. Lloyd, Quantum Algorithm Providing Exponential Speed Increase for Finding Eigenvalues and Eigenvectors, Physical Review Letters, vol. 83, no. 24, pp. 51625165, Dec. 1999.
- [19] B. T. Torosov and N. V. Vitanov, Design of quantum Fourier transforms and quantum algorithms by using circulant Hamiltonians, Physical Review A, vol. 80, no. 2, Aug. 2009.
- [20] J. S. Lee, J. Kim, Y. Cheong, and S. Lee, Implementation of phase estimation and quantum counting algorithms on an NMR quantum-information processor, Physical Review A, vol. 66, no. 4, Oct. 2002.
- [21] C. Travaglione and G. J. Milburn, Generation of eigenstates using the phase-estimation algorithm, Physical Review A, vol. 63, no. 3, Feb. 2001.
- [22] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Quantum algorithms revisited, Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, vol. 454, no. 1969, pp. 339354, Jan. 1998.
- [23] S. Lloyd, M. Mohseni, and P. Rebentrost, Quantum principal component analysis, Nature Physics, vol. 10, no. 9, pp. 631633, Jul. 2014.
- [24] Rebentrost, Patrick and Mohseni, Masoud and Lloyd, Seth Quantum Support Vector Machine for Big Data Classification Phys. Rev. Lett., vol. 113, pp. 130503, sep. 2014.