Learning Quantum Phase Estimation by Variational Quantum Circuits

Team *LearnQPE*

Motivation

Quantum Phase Estimation (QPE) has low performance on NISQ hardware due to accumulated gate errors, yet QPE is crucial in many algorithms. Thus, we aim to mimic QPE result.

01

Quantum Phase Estimation

Purpose, Process & Problem

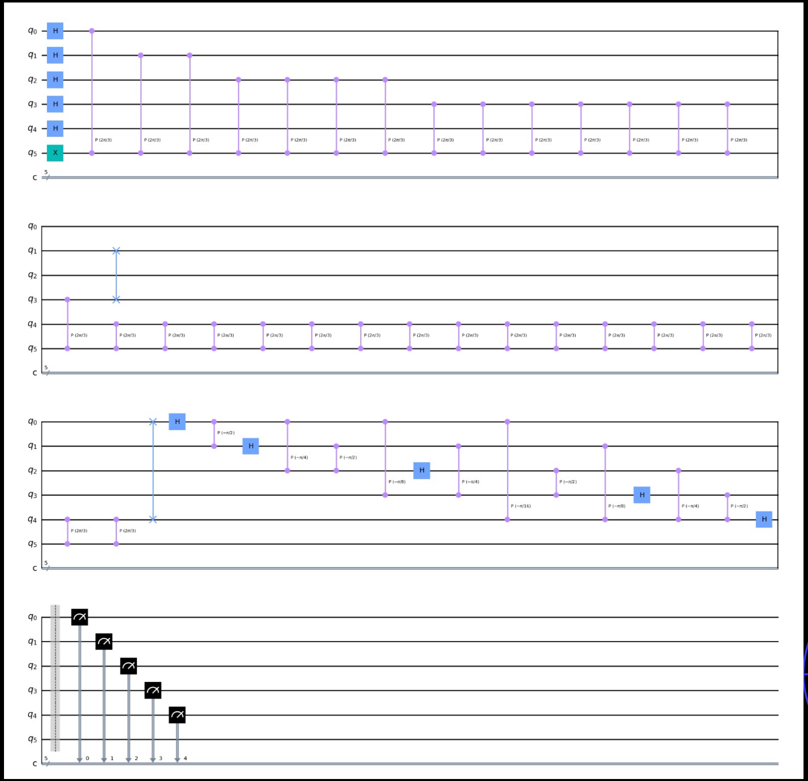
Purpose of QPE

- Estimate the θ in $e^{2\pi i\theta}$, the eigenvalue of an eigenvector of an unitary operator U
- Simplify complex problems

→ An important subroutines in many quantum algorithms and computation

Process of QPE

- Superposition, Controlled unitary operations, and Inversed Fourier Transformation.



Problem with QPE

- Increasing the accuracy (the number of first register) increases the circuit depth in geometric order
- Accumulated gate errors on NISQ hardware cause undesired result

02

Variational Quantum Circuit

Components, Cost function, Entangler

Components of VQC

$|0\rangle$

$U(\theta)$

Initial State

Prepare initial states of $|0\rangle$

Parameterized Quantum Circuit

Consist of adjustable gates $U(\theta)$, θ: the parameters trained by classical optimizer

Measurement

Feed measurements, (a.k.a. expectation values) to classical optimizer

Classical Optimizer & Cost Function

- Cost values is defined as the mean square error of the measurement result between QPE circuit and VQC
- Classical Optimizer will update the parameters to minimize the cost value
- Classical optimizer will make post-measurement calculation of the cost value using the measurement result
- Thereby obtain the set of parameters that produce the desired approximation

Full & Linear Entangler

Full

Create entanglement of all combinations of qubits
→ More Expressive
→ has significant difference when there are more qubits

Linear

Create entanglement of only the adjcent qubits
→ Less gates
→ Comparable when there are less qubits

Pierpaolo Pravatto *et al* 2021 *New J. Phys.* **23** 123045

03

Learning QPE by VQC

Applicability, Method

LearnQPE process

01

Run QPE simulation

Obtain the ideal wave function after QPE

02

Train parameters

Use the simulation result & optimizer to train the free parameters

03

Feed back to VQC

Use the trained parameters to set the gates in VQC

04

Run trained VQC

Run the VQC; yield the approximated wave function

Applicability

QPE Simulation

Must run the QPE simulation to obtain a ground truth to train the VQC

Simulation of QPE on classical computer must be achievable for this approach to function.

In this project, we use the QPE circuit with 5 counting qubits as the example in the Qiskit documentation

Our Method

Learn QPE

5 counting qubits

Yield more accurate approximation for the phase angle

R_yR_z Form

Layer = 1, one set of entangler & layer of R_yR_z rotations

Full Entangler

More epressive

COBYLA

Only one function evaluation per iteration

Testing Condition

01

Ideal QPE

QPE simulation without noise
(backend = ibmq_qasm_simulator)

02

Noisy QPE

QPE simulation with noise
(backend= ibm_oslo_sim)

03

QPE real hardware

Run QPE circuit on ibm_oslo

04

Ideal VQC

VQC simulation without noise
(backend = ibmq_qasm_simulator)

05

Noisy VQC

VQC simulation with noise
(backend= ibm_oslo_sim)

06

VQC real hardware

Run VQC circuit on ibm_oslo
(Train with noisy simulation)

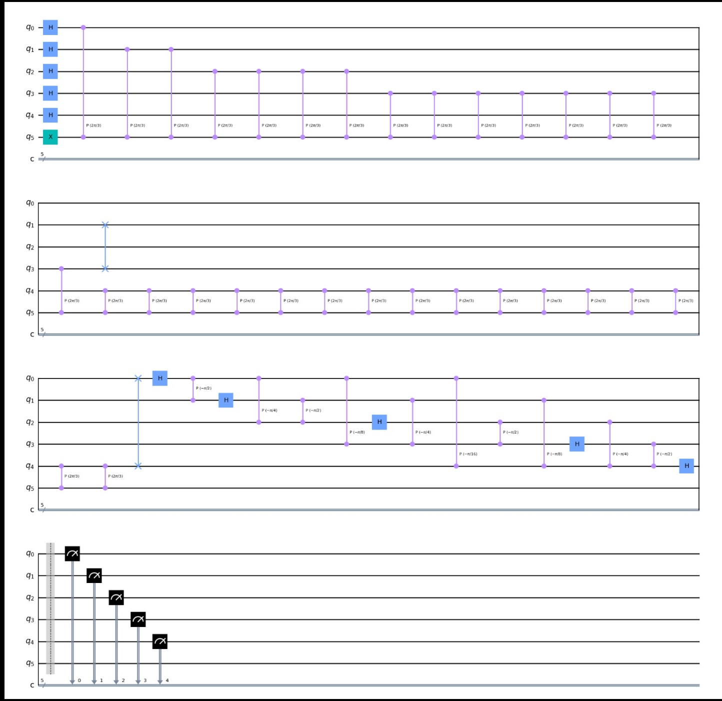
04

Results

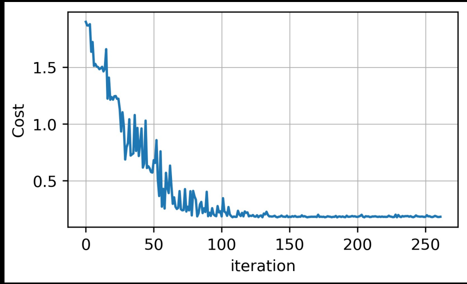
Comparison between VQC & QPE

Results

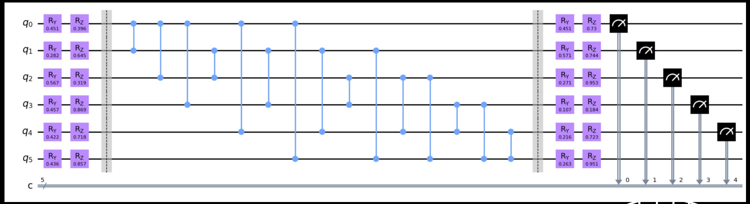
Original QPE circuit (depth : 43)



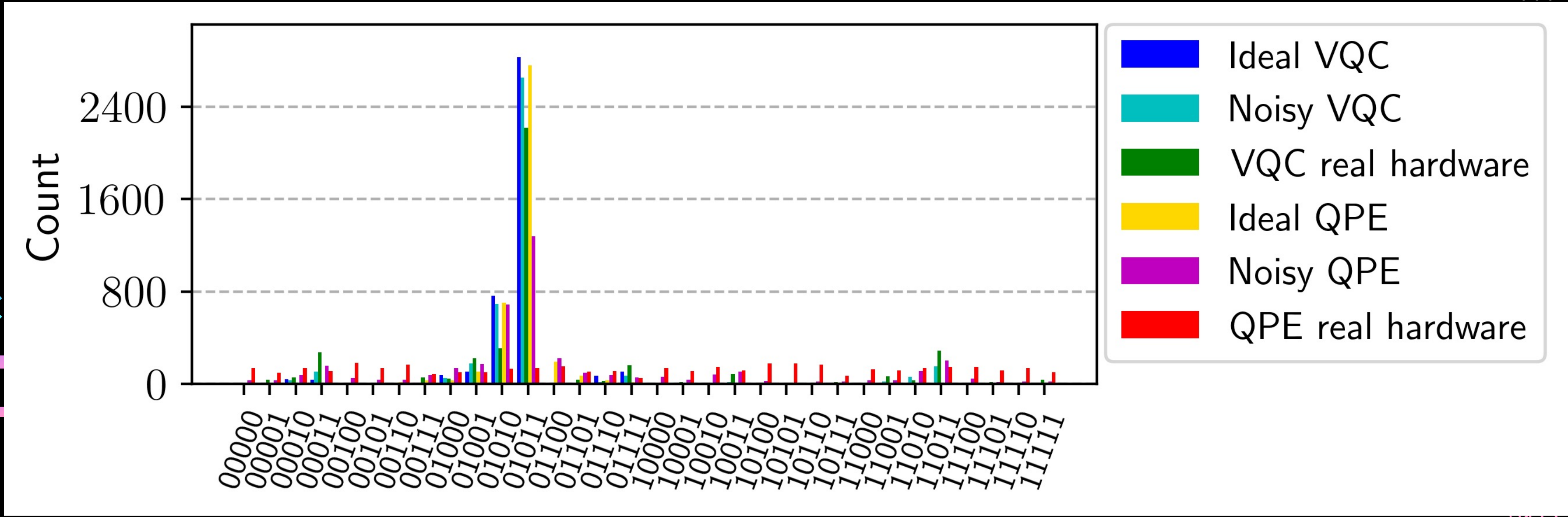
Learning



Corresponding VQC (depth : 14)



Results


Count
2400
1600
800
0
Ideal VQC
Noisy VQC
VQC real hardware
Ideal QPE
Noisy QPE
QPE real hardware
00000
00001
00010
00011
00100
00101
00110
00111
01000
01001
01010
01011
01100
01101
01110
01111
10000
10001
10010
10011
10100
10101
10110
10111
11000
11001
11010
11011
11100
11101
11110
11111

Result Analysis

- Ideal VQC output a very similar result to that of Ideal QPE
- Noisy VQC performs better than Noisy QPE, compatible performance to that of Ideal VQC
- VQC real hardware has performs better than QPE real hardware, even better performance than that of Noisy QPE

05

Discussion & Conclusion

As Part of the Quantum Complier

01

Inputted QPE circuit

User inputted quantum circuit

02

Classical simulatable

If the circuit is simulatable, we can construct a corresponding VQC to approximate the result

03

Input corresponding VQC

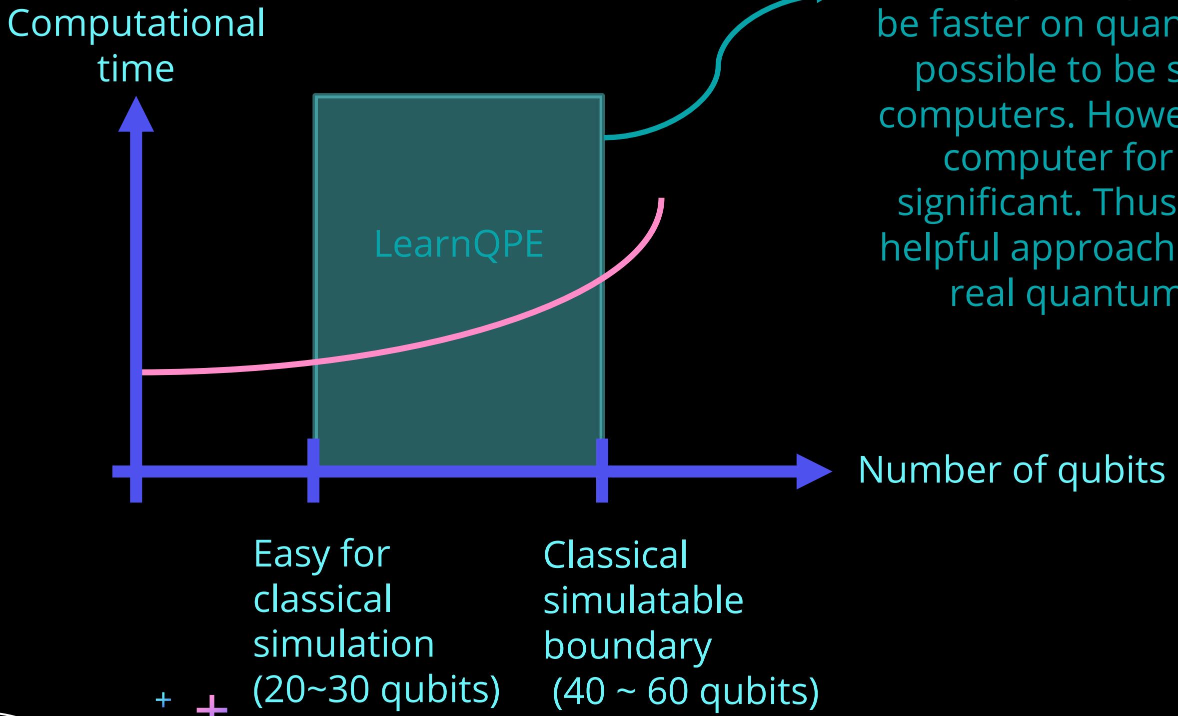
The corresponding VQC usually have shallower circuit depth, which reduce the noise significantly

04

Run VQC

Run the VQC; yield the approximated wave function on the quantum hardware

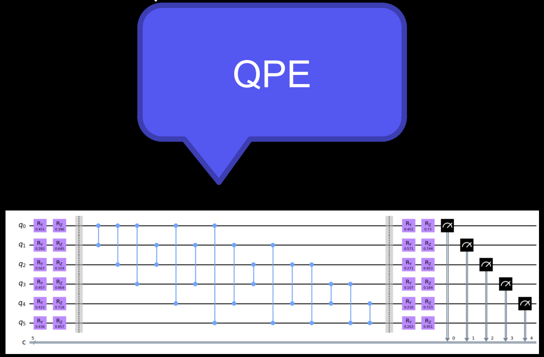
Helpful Use Case

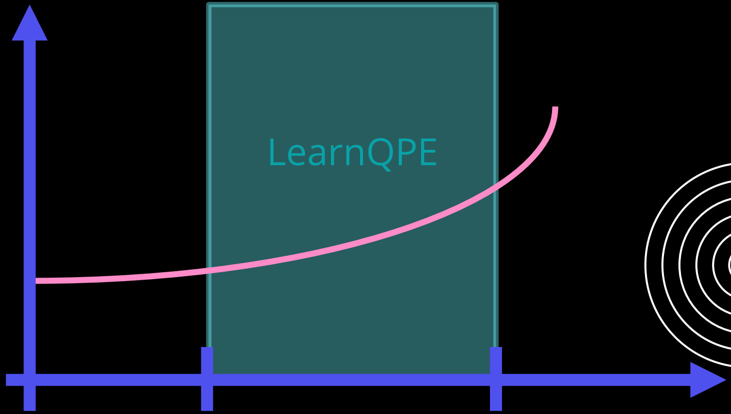


In this regime, quantum circuit executions may be faster on quantum computers, but is also possible to be simulated on the classical computers. However, the noise on quantum computer for this scale could also be significant. Thus, our LearnQPE could be a helpful approach to reduce the noise on the real quantum hardware in this case.

Conclusion

- Use VQC to simulate the QPE circuit.
- Noise of quantum hardware is mitigated by the reduction of circuit depth.
- This method can be generalized to other quantum algorithms that suffer from deep quantum circuit.





Thanks!