2025 Quantum Hackathon

Classification

Quantum Algorithms: Quantum Phase Estimation

Background

Quantum Phase Estimation (QPE) is a fundamental quantum algorithm that estimates the eigenvalue (or phase) corresponding to an eigenvector of a given unitary operator. It plays a crucial role in various quantum applications, including Shor's factoring algorithm, quantum simulations, and quantum chemistry. QPE encodes phase information into a quantum register using controlled unitary operations and extracts it through the inverse quantum Fourier transform. Although the algorithm is conceptually straightforward, its practical implementation requires translating the mathematical descriptions of the unitary operator and input state into executable quantum circuits, which is often a challenging task in practice.

Problem

1. QPE Implementation

Given the unitary operator $U = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i\theta} \end{bmatrix}$ and the eigenvector $v_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, implement the QPE algorithm to estimate the phase θ in the following cases:

1.1. $\theta = 5/8$

1.2. $\theta = 3/7$

For each case, analyze how increasing the number of qubits used for the estimation affects the precision of the output.

2. Finding Eigenvalues of a Matrix using QPE

Given the matrix

$$M = \begin{bmatrix} 1 & 0 & 8 & 1 \\ 0 & 1 & 1 & 8 \\ 8 & 1 & 1 & 0 \\ 1 & 8 & 0 & 1 \end{bmatrix},$$

estimate all eigenvalues of M using only the QPE algorithm. You are not allowed to use classical methods to compute eigenvectors in advance.

Note 1: Your algorithm should be designed to handle the case where the matrix is not positive definite.

Note 2: The spectral norm can be upper bounded using either the 1-norm or the ∞-norm.

Note 3: While you may use built-in functions to simulate matrix exponential, higher scores will be awarded for implementations that explicitly decompose M into quantum operators and manually construct the corresponding circuit.

Note 4: You are encouraged to estimate the eigenvalues without relying on prior knowledge of the eigenvectors. If not, your submission may receive a lower score.

3. Analyzing the Impact of Noise on QPE

QPE is known to be sensitive to various types of quantum noise. Using your implementation from the previous problems, apply different noise models (e.g., depolarizing noise, readout error) and analyze how the accuracy of QPE is affected.

4. Exploring Applications of QPE

Identify an application where the QPE algorithm can be applied effectively. Construct a simple example that demonstrates the use of QPE in that application, and explain how the algorithm contributes to solving the problem.

References

[1] M. A. Nielsen and I. L. Chuang, Quantum computation and quantum information (Springer, 1999)

[2] Thomas G. Wong, Introduction to classical and quantum computing (Rooted Grove, 2022)

[3] L. Lin, Lecture Notes on Quantum Algorithms for Scientific Computation (https://arxiv.org/abs/2201.08309)