Exploring Variations of Quantum Phase Estimation Algorithm

iQuHACK 2024 – Quantinuum Challenge

Emma Yang[1] Ricardo Skewes[1] William Hu[1] Ervin Grimaldi[2]

[1]Harvard University. [2]University of California, Irvine

February 4, 2024

Table of Contents

Introduction

- Quantum Phase Estimation (QPE) is a fundamental algorithm in quantum computing, crucial for applications in quantum chemistry, cryptography, and quantum simulations, and is a subroutine of well-known algorithms such as Shor's algorithm and the quantum algorithm for solving linear systems.
- The accuracy and efficiency of QPE directly impact the performance of these quantum applications.
- Our focus: Exploring variations of QPE and evaluating their precision and performance.

Why Study Variations of QPE?

- The standard QPE algorithm, while powerful, requires a significant number of qubits and quantum operations, which may introduce errors and increase computational cost.
- By studying variations, we aim to:
  1. Reduce the number of required qubits and operations.
  2. Increase the algorithm's robustness to errors.
  3. Adapt the algorithm for near-term quantum computers.

Quantum Phase Estimation (QPE)

- The QPE algorithm estimates the phase (ϕ) of an eigenvalue of a unitary operator U, where $U|\psi\rangle = e^{2\pi i\phi}|\psi\rangle$.
- Key components:
  1. A register of m qubits for phase estimation.
  2. Controlled-U operations to encode phase information.
- The algorithm's precision depends on the number of qubits (m) and the number of shots (number of circuit executions over which the probability distribution of outcomes is gathered).

Experiment Setup

- Conducted experiments with the vanilla version of QPE while varying the **number of shots**, **number of qubits**
- Each experiment consisted of testing 10 initial phases, sampled uniformly at random from $\theta \in [0, 1]$.
- Varied the number of precision qubits (m) from 2 to 10 to study the impact on the algorithm's precision and error rates.
- Tested a range of shots evenly sampled from 15 to 1500 shots.

Experiment Setup

- Experiments were conducted as a grid search in uniformly sampled at random from each search space

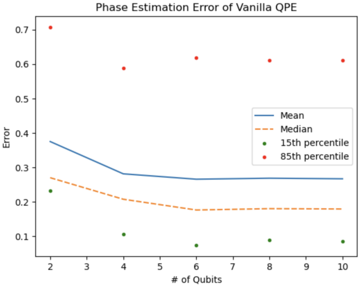
  $\theta \in [0, 1]$ Number of Qubits $\in [2, 12]$ Number of Shots $\in [15, 1500]$

- For each of the parameters we were evaluating, the QPE circuit was initialized with the trivial $|1\rangle$ eigenstate for the U1 unitary gate and each of the theta angles
- Since each simulation could take up to 15 minutes to run, it was important that we streamline our experimental framework and maximize the volume of results we could have to analyze
- Circuit compilation and evaluation was batched in sets of 10 circuits at a time to the Nexus platform for maximum parallelization
- Results were submitted/queued on Quantinuum devices asynchronously, then polled by a Python program which aggregated results as they became available for analysis
- Iterative QPE circuits were evaluated with the offline Quantinuum API

Vanilla QPE: Error Analysis

- Varying the number of qubits (circuit depth) while keeping the number of shots constant at 1,000 shots empirically verifies that increasing the number of qubits decreases the error of phase estimation.
- The data also shows that the estimation is prone to relatively large right tails in the error distribution, at this level of θ-sampling in experimentation
- The local minimum for the error was achieved at 6 qubits but the error is not significantly different from neighboring numbers of qubits in the plot, suggesting that larger numbers of experiments (100-1K random θ phase measurements) would improve our understanding of this landscape.

Experimental Results



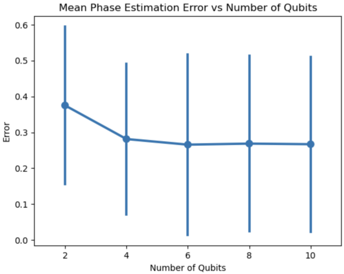
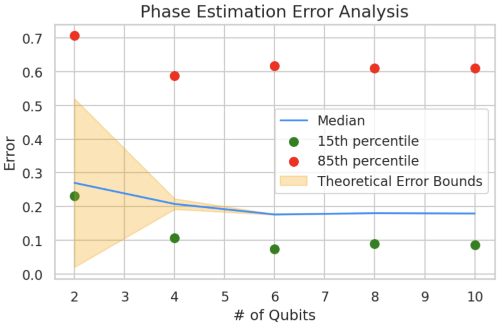


Figure: Mean Phase Estimation Error $\pm$ 1 Standard Deviation

Experimental Results



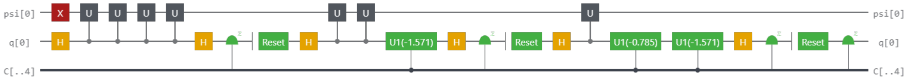
Theoretical error bounds

$$\left|\frac{1}{2m}\sum_{k=0}^{2m-1} e^{ik(\theta-2\pi j)/2m}\right|^2 = \left|\frac{1}{4m}\frac{1-e^{i(2L\theta-2\pi j)}}{1-e^{i(\theta-2\pi j)/2m}}\right|^2.$$

$$\left|\frac{1}{4m}\right|\left|\frac{2}{2\pi(j'-j)/2m}\right|^2 \leq \frac{1}{|\pi\alpha|^2}.$$

Variation 1: Iterative QPE

- Iterative QPE reduces the number of qubits by using a single qubit repeatedly to estimate the phase bit by bit.
- Instead of using rotations controlled on qubits in the inverse QFT, control on the previous measurements.
  1. A register of 1 qubit
  2. A classical register of m bits
  3. Controlled-U operations to encode phase, conditioned now on the previous measurements
- This variation is particularly useful for quantum systems with limited qubit availability.
- **Advantages:** Requires fewer qubits and can be more resilient to certain types of errors.



Variation 2: Bayesian QPE

- Bayesian QPE uses Bayesian inference to update the probability distribution of the phase estimate based on measurement outcomes.
- It can provide accurate phase estimates with fewer measurements by intelligently choosing the next measurement basis.
- **Advantages:** Higher precision with fewer shots and adaptability to prior information.

Variation 2: Bayesian QPE

Framework:

- Obtain an estimate of the distribution of θ, then can estimate $\mathbb{E}[\theta]$ and $\mathrm{Var}(\theta)$.
- Perform Bayesian update on samples drawn from a Gaussian for each digit
- Perform experiment for given θ, M, and observe outcome $E \in \{0, 1\}$.
- Draw m samples from $\mathcal{N}(\theta \mid \mu, \sigma^2)$
- For each sample, assign θ_j.

Conclusion

- Our exploration into variations of the Quantum Phase Estimation algorithm empirically verifies theoretical bounds on the phase estimation error with experiments run on Quantinuum's Nexus platform
- Our codebase demonstrates a generalizeable experimental harness for conducting larger scale experiments in parallel with the Nexus API and devices which can be extended to running parallel simulations, given a function that generates a circuit and a set of hyperparameters
- Future work includes implementing and testing Bayesian Phase Error Estimation with reject sampling and showing how improvements on QPE can contribute to minimizing error in algorithms where QPE is a subroutine.

Questions?

Thank you for your attention!
Any questions?