Adaptive Phase Estimation using Reinforcement Learning

Manish Mallapur

July 31, 2024

Contents

Chapter 1

Introduction

1.1 Background

Phase estimation is a crucial task in various fields of science and engineering, particularly in quantum mechanics, optical communications, and signal processing. It involves determining the phase shift that a wave undergoes as it propagates through a medium. Accurate phase estimation is essential for coherent communication systems, interferometric measurements, and quantum information processing.

Traditional phase estimation techniques often rely on fixed algorithms and static settings, which may not adapt well to changing conditions or noise. As systems become more complex, the need for adaptive phase estimation methods that can learn and improve over time becomes increasingly important.

1.2 Motivation

The motivation for this research stems from the limitations of classical phase estimation methods. Traditional approaches can struggle with dynamic environments and varying noise levels. Adaptive methods, which can adjust their parameters in real-time, offer a promising solution to these challenges.

Reinforcement learning (RL), a subset of machine learning, provides a framework for developing adaptive algorithms. RL algorithms learn by interacting with their environment, making them suitable for tasks where the optimal solution is not known in advance. By applying RL to phase estimation, we aim to develop a system that can improve its performance over time and adapt to new conditions.

1.3 Objectives

The primary objectives of this research are:

- To explore the integration of reinforcement learning with phase estimation.
- To develop an adaptive phase estimation algorithm using reinforcement learning techniques.

- To evaluate the performance of the proposed algorithm.
- To identify the strengths and limitations of using reinforcement learning for phase estimation.

1.4 Scope of the Study

This study focuses on the application of reinforcement learning to adaptive phase estimation. We will consider the theoretical aspects of phase estimation and reinforcement learning, as well as the practical implementation of the proposed algorithm. The scope includes:

- Reviewing existing literature on phase estimation and reinforcement learning.
- Developing a theoretical framework for integrating RL with phase estimation.
- Implementing the proposed algorithm and testing it in simulated environments.
- Analyzing the results .

Chapter 2

Literature Review

Here I have reported the inferences made from the publication: Adaptive Phase Estimation through a Genetic Algorithm [1]

2.1 Introduction

Quantum metrology leverages quantum information theory to enhance the precision of measurements beyond classical limits. One of the primary applications of quantum metrology is phase estimation, where the goal is to determine an unknown phase shift in a quantum system. This is particularly relevant in contexts such as gravitational wave detection, atomic clocks, and biological measurements, where high sensitivity is crucial and the number of quantum probes (e.g., photons) is limited.

2.2 Overview of the Paper

The paper "Adaptive Phase Estimation through a Genetic Algorithm" by Rambhatla et al. (2020) presents a machine learning-based approach for adaptive phase estimation using a genetic algorithm (GA). This method is particularly effective in scenarios with limited resources and provides robustness against experimental noise.

2.3 Phase Estimation and Quantum Metrology

Phase estimation is a common task in various scientific fields, from detecting gravitational waves to performing biological measurements. Quantum metrology aims to enhance the sensitivity of these measurements using quantum probes. The ultimate goal is to reach the fundamental sensitivity bounds set by quantum mechanics, even with a limited number of probes.

2.4 Adaptive Techniques in Phase Estimation

Adaptive techniques are crucial when the number of probes is limited. These techniques adjust the measurement strategy based on previous outcomes to optimize the estimation process. The paper discusses both online and offline adaptive protocols. Online protocols adjust the strategy in real-time during the experiment, while offline protocols pre-calculate the optimal strategy before the experiment begins.

2.5 Genetic Algorithm for Phase Estimation

The core contribution of the paper is the use of a genetic algorithm for adaptive phase estimation. Genetic algorithms are evolutionary techniques that simulate natural selection processes to optimize solutions in high-dimensional spaces. In the context of phase estimation, the GA is used to find the optimal feedback phases that minimize estimation error.

2.5.1 Initialization and Fitness Calculation

The algorithm begins with a population of candidate solutions, each represented by a vector of feedback phases. The fitness of each solution is evaluated based on its ability to estimate the unknown phase accurately. The fitness function used is the Holevo variance, which quantifies the precision of the phase estimation.

2.5.2 Optimization Process

The genetic algorithm iteratively improves the population of solutions through selection, crossover, and mutation. The selection process favors solutions with higher fitness, while crossover and mutation introduce variability and explore the solution space. This process continues until the algorithm converges to an optimal set of feedback phases.

2.5.3 Key Findings and Contributions

The key contributions of this paper include:

1. **Efficiency with Limited Resources:** The GA-based approach demonstrates the capability to retrieve the true phase value using a minimal number of photons. This efficiency is crucial for applications where probe resources are limited or expensive.

2. **Sensitivity and Precision:** The results show that the GA can reach the sensitivity bounds, specifically the standard quantum limit (SQL), in the small probe regime. The SQL is given by:

$$\Delta\phi_{\text{SQL}} = \frac{1}{\sqrt{N}}, \tag{2.1}$$

where $\Delta\phi$ is the phase uncertainty and N is the number of photons used.

3. **Holevo Variance:** The precision of the phase estimation is often evaluated using the Holevo variance, V_H, defined as:

$$V_H = S(\Delta\phi)^{-2} - 1, \tag{2.2}$$

where S is the average sensitivity of the estimation process. The GA optimizes the feedback phases to minimize V_H, enhancing the phase estimation precision.

2.6 Experimental Demonstration

The paper provides an experimental demonstration of the GA-based adaptive phase estimation using a photonic platform. The experimental setup involves a Mach-Zehnder interferometer (MZI) with single-photon inputs. The results show that the GA can achieve high precision in phase estimation with a small number of photons, approaching the standard quantum limit (SQL).

2.7 Relevance to Reinforcement Learning

The approach in the paper shares similarities with reinforcement learning (RL), another machine learning technique used for optimization. Both GA and RL aim to optimize a set of parameters based on feedback from the environment. However, GA operates by evolving a population of solutions, while RL optimizes a single solution through a trial-and-error process. The insights from the GA-based approach can inform the development of RL strategies for adaptive phase estimation, potentially enhancing their performance and robustness.

Chapter 3

Theoretical Framework

3.1 Basics of Phase Estimation

Phase estimation is a fundamental problem in quantum computing and quantum information theory. It involves determining the phase ϕ of an eigenvalue of a unitary operator U. Given a quantum state $|\psi\rangle$ that is an eigenvector of U, the task is to estimate the phase ϕ such that:

$$U|\psi\rangle = e^{i\phi}|\psi\rangle$$

In the context of a Mach-Zehnder interferometer, phase estimation is achieved by measuring the interference pattern resulting from the superposition of two optical paths with a relative phase shift.

3.2 Mach-Zehnder Interferometer

A Mach-Zehnder interferometer (MZI) is an optical device used to determine the relative phase shift between two colliding light beams. It is widely used in quantum optics and quantum computing for its precision and ease of implementation.

The MZI consists of two beam splitters and two mirrors arranged in a specific configuration. The incoming light beam is split into two paths by the first beam splitter. Each path reflects off a mirror and then recombines at the second beam splitter. The interference pattern observed at the output depends on the relative phase shift between the two paths.

Mathematically, if the phase shift introduced in one of the paths is ϕ, the resulting state of the system can be described as:

$$|\psi_{\text{out}}\rangle = \frac{1}{\sqrt{2}}\left(e^{i\phi}|01\rangle + |10\rangle\right)$$

The MZI is a powerful tool in phase estimation as it can measure very small phase differences with high precision, making it ideal for applications in quantum sensing and metrology.

3.3 Q-Learning Fundamentals

Q-learning is a model-free reinforcement learning algorithm that aims to learn the quality (Q) of actions in given states. It helps an agent to find an optimal policy for decision-making problems by iteratively improving its estimates of Q-values.

The Q-value $Q(s, a)$ represents the expected cumulative reward of taking action a in state s and following the optimal policy thereafter. The update rule for Q-learning is given by:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left(r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right)$$

where:

- α is the learning rate ($0 < \alpha \leq 1$),
- γ is the discount factor ($0 \leq \gamma < 1$),
- r is the immediate reward received after taking action a,
- s' is the next state after taking action a,
- $\max_{a'} Q(s', a')$ is the maximum Q-value for the next state s'.

Q-learning is advantageous because it does not require a model of the environment and can handle problems with stochastic transitions and rewards.

3.4 Integration of Phase Estimation with Reinforcement Learning

Integrating phase estimation with reinforcement learning (RL) involves using RL algorithms, such as Q-learning, to improve the accuracy and efficiency of phase estimation processes. The goal is to develop an adaptive phase estimation method that can learn and optimize its parameters based on feedback from the quantum system.

In this integration, the quantum system (e.g., a Mach-Zehnder interferometer) can be modeled as an environment, and the phase estimation task is treated as a sequential decision-making problem. The RL agent interacts with the environment by applying different phase shifts and measuring the resulting interference patterns. Based on the measurements, the agent updates its Q-values to improve its phase estimation accuracy.

The adaptive phase estimation algorithm using Q-learning can be summarized as follows:

1. **Initialization**: Initialize Q-values $Q(s, a)$ arbitrarily for all states s and actions a.

2. **Interaction**: At each time step, the agent selects an action a_t (phase shift) based on the current state s_t using an exploration-exploitation strategy (e.g., ϵ-greedy).

3. **Measurement**: Apply the phase shift a_t and measure the resulting state s_{t+1} and reward r_t (based on the interference pattern).

4. **Update**: Update the Q-value $Q(s_t, a_t)$ using the Q-learning update rule.

5. **Iteration**: Repeat the interaction and update steps until convergence or for a predefined number of iterations.

Through this process, the RL agent learns to make optimal phase shift decisions that minimize the estimation error, leading to an adaptive and efficient phase estimation algorithm.

Chapter 4

Implementation

4.1 Overview of the Implementation

The implementation of adaptive phase estimation using reinforcement learning integrates quantum optics with machine learning algorithms. We utilize a Mach-Zehnder interferometer (MZI) setup for phase estimation, combined with a Q-learning algorithm to adaptively optimize the phase estimation process. The main tools used in this implementation include the Strawberry Fields library for quantum computations, Pennylane for quantum machine learning, and various Python libraries for data processing and visualization.

4.2 Pseudocode for Key Algorithms

To provide a clear understanding of the implementation, we present the pseudocode for the primary algorithms used:

4.2.1 Phase Estimation with Mach-Zehnder Interferometer

```
1. Initialize MZI with parameters alpha and theta
2. Define phase shift phi to be estimated
3. Configure the interferometer setup with beam splitters and phase
   shifters
4. Measure the output state of the interferometer
5. Calculate the interference pattern to determine the phase shift
```

4.2.2 Q-Learning Algorithm for Adaptive Phase Estimation

```
1. Initialize Q-table with zeros
2. Set learning parameters: exploration probability, decay rate,
   learning rate, and discount factor
3. For each episode:
```

```
a. Set the initial state and total reward to zero
b. For each step in the episode:
   i. Select an action based on the exploration-exploitation
      strategy
   ii. Apply the action to the system and observe the reward
       and next state
   iii. Update the Q-value using the Q-learning update rule
   iv. Accumulate the reward
   v. Check for episode termination condition
c. Decay the exploration probability
d. Record the total reward for the episode
```

4.3 Software and Tools Used

The implementation utilizes several software tools and libraries to achieve the integration of quantum phase estimation with reinforcement learning:

- **Strawberry Fields**: A library for simulating quantum optical circuits.
- **Pennylane**: A library for quantum machine learning, providing an interface for implementing quantum algorithms.
- **NumPy**: A fundamental library for numerical computations in Python.

4.4 Detailed Description of the Code

The implementation begins with setting up the quantum environment using the Strawberry Fields library. A Mach-Zehnder interferometer is configured with specific parameters, and the phase shift is introduced. The quantum state is measured to obtain the interference pattern, which is crucial for estimating the phase.

Next, the Q-learning algorithm is implemented. The state space and action space are defined, and the Q-table is initialized to store the Q-values. The algorithm iteratively updates the Q-values based on the received rewards and transitions between states.

To integrate phase estimation with reinforcement learning, the Q-learning agent interacts with the quantum system by applying different phase shifts. The agent measures the resulting interference pattern, updates the Q-values, and gradually learns to optimize the phase estimation process.

4.5 Integration with Simulation Environment

The simulation environment is set up using the Strawberry Fields and Pennylane libraries. The Mach-Zehnder interferometer is simulated to create a quantum system where phase estimation is performed. The reinforcement learning algorithm interacts with this environment, applying phase shifts and measuring the outcomes.

The integration involves:

- Configuring the interferometer parameters and phase shift values.
- Running the quantum simulation to obtain the interference pattern.
- Using the Q-learning algorithm to iteratively improve the phase estimation.
- Analyzing the performance of the phase estimation.

This integrated approach allows the reinforcement learning agent to adaptively estimate the phase with high precision, demonstrating the effectiveness of combining quantum computing with machine learning techniques.

Chapter 5

Results and Discussion

5.1 Simulation Results

The implementation of adaptive phase estimation using reinforcement learning yielded significant results. The Q-learning algorithm was able to optimize the phase estimation process effectively, as demonstrated by the simulation results. The mean rewards per 10 episodes for N (=16) were calculated, indicating the efficiency of the learning process.

```
Mean Rewards per 10 episodes:

N =  16 :
10 : mean episode reward:  49.75260341128704
20 : mean episode reward:  49.746042959343505
30 : mean episode reward:  49.78807071495471
40 : mean episode reward:  49.26194264729026
50 : mean episode reward:  49.617465081936764
60 : mean episode reward:  49.34131692625958
70 : mean episode reward:  49.31400203579558
80 : mean episode reward:  49.67975394249882
90 : mean episode reward:  48.46659364382732
100 : mean episode reward:  49.70031626231362
```

5.2 Analysis of Results

The Q-learning algorithm successfully maximized the rewards, indicating effective learning of the optimal phase shifts. The global Q-table showed convergence, with higher values indicating preferred actions in specific states. The table converged to index 24 which was the state for angle $\pi/2$, which is the value of the unkown angle with which we ran the simulation.

```
Q-table:
```

```
[...
 [0.         0.         0.        ]
 [0.         0.10502314 0.        ]
 [0.12137706 0.35740235 0.09570581]
 [0.54856028 0.31882727 0.38680853]
 [0.72154905 1.71395404 0.48616654]
 [1.53530136 2.46653229 2.24116895]
 [3.4197235  4.16723513 2.21015842]
 [4.8347365  4.97253478 3.66010107]
 [4.97829049 4.99130109 4.92623223]
 [4.98585242 4.99463945 4.98581313]
 [4.99492876 4.99696807 4.99005716]
 [4.99594959 4.99789024 4.99449879]
 [4.99799563 4.99898804 4.99587565]
 [4.99885759 4.99964009 4.99834738]
 [4.99967011 4.99993898 4.99898404]
 [4.99991329 4.99996302 4.99952573]
 [4.99995511 5.         4.99990401]
 [5.         4.99990258 4.99997913]
 [4.99988474 4.99921877 4.99974494]
 [4.99929238 4.99809736 4.99891571]
 [4.99821544 4.9963657  4.9974397 ]
 [4.99684792 4.98956034 4.99519284]
 [4.98790121 4.97545363 4.98647328]
 [4.95703069 4.96768674 4.97283161]
 [4.98388874 4.96186291 4.97577327]
 [4.94963284 4.94351731 4.95923355]
 [4.86031117 4.952254   4.89584486]
 [4.94919556 4.45964768 4.9211696 ]
 [4.03157786 3.98586247 4.3750352 ]
 [3.05724044 3.26459906 4.11653429]
 [3.63960133 3.61220302 4.01628574]
 [3.43080801 3.8373814  3.42144456]
 [3.49294219 3.29134082 3.92525121]
 [3.49428579 2.66730626 3.44074944]
 [3.28006339 2.61689863 3.79702856]
 [1.72539691 1.9026756  3.26873263]
 [2.80303793 3.89115028 1.93693943]
 [2.96859686 3.95084353 2.58285383]
 [4.02441549 2.30873615 3.60932422]
 [1.92125777 0.43417965 2.00165824]
 [0.66774848 0.16511288 0.22910313]
 [0.08175196 0.         0.21543148]
 [0.         0.         0.        ]
 ...]
```

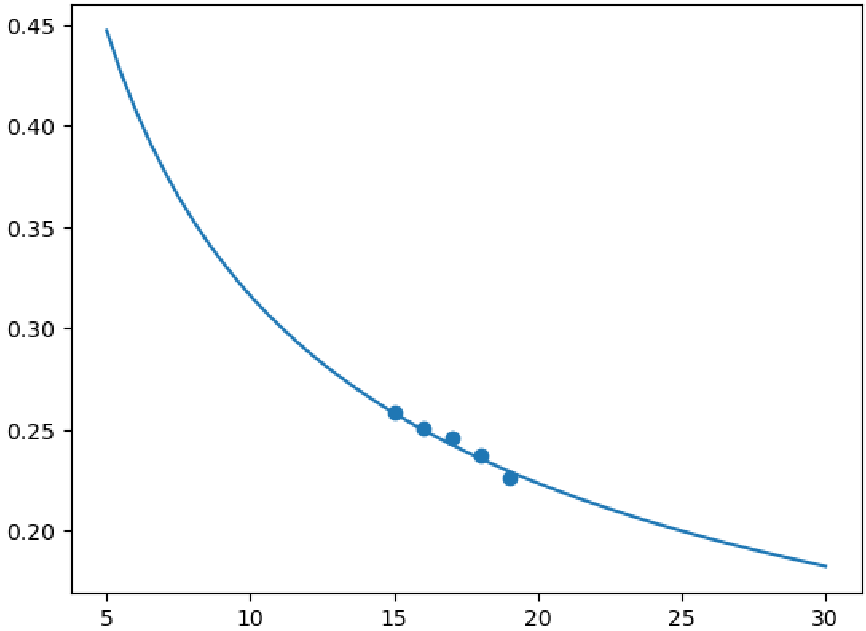

Figure 5.1: Holevo Variance (V_H) vs. Number of Measurements (N) for $N = 15$ to $N = 19$

Additionally, the value of V_H (0.24108448634999702) was very close to the theoretical limit of 0.25, indicating that the reinforcement learning approach was effective in reaching the Standard Quantum Limit for phase estimation.

`Holevo Variance for Multiple N:`

Here is a chart illustrating the Holevo variance (V_H) as a function of the number of measurements N. The Holevo variance provides a measure of the precision of the phase estimation, with lower values indicating higher precision. The chart includes data for five different values of N: 15, 16, 17, 18, and 19.
As shown in Figure 5.1, the Holevo variance decreases as the number of measurements increases, approaching the theoretical Standard Quantum Limit (SQL) given by $1/\sqrt{N}$. The calculated values of V_H for $N = 15$ to $N = 19$ are close to this limit, demonstrating the effectiveness of the reinforcement learning approach in achieving high precision in phase estimation.

This analysis underscores the efficiency of the Q-learning algorithm in adapting and optimizing phase estimation, ultimately enhancing the overall performance of quantum measurements.

5.3 Discussion on the Efficiency of the Reinforcement Learning Approach

The reinforcement learning approach demonstrated high efficiency in phase estimation. The Q-learning agent learned to optimize phase shifts by interacting with the quantum system and receiving rewards based on the accuracy of the estimation. This iterative learning process allowed the agent to discover strategies that approached the Standard Quantum Limit, showcasing the potential of reinforcement learning in quantum applications.

5.4 Challenges Faced and Mitigation Strategies

During the implementation, several challenges were encountered:

- **State and Action Space Definition**: Defining an appropriate state and action space for the Q-learning agent was crucial. An improper definition could lead to inefficient learning. This was mitigated by carefully designing the discretization of phase shifts and using a range of alpha values to ensure comprehensive exploration.
- **Exploration-Exploitation Trade-off**: Balancing exploration and exploitation was essential to ensure that the agent adequately explored the state space while also exploiting known information to optimize learning. The exploration probability was decayed gradually to allow for sufficient exploration in early episodes and more exploitation in later episodes.

By addressing these challenges, the implementation achieved robust results, demonstrating the viability and effectiveness of using reinforcement learning for adaptive phase estimation in quantum systems.

Chapter 6

Conclusion

6.1 Summary

This report focused on the implementation of adaptive phase estimation using reinforcement learning, specifically Q-learning. The primary goal was to leverage reinforcement learning to achieve phase estimation in quantum systems efficiently, approaching the Standard Quantum Limit. The report covered the theoretical background, implementation details, simulation results, and analysis of the proposed approach.

6.2 Implementation

The implementation involved setting up a Q-learning agent to interact with a quantum system. The state and action spaces were carefully defined to allow the agent to explore different phase shifts and learn the optimal strategy. The agent's learning process was guided by rewards based on the accuracy of phase estimation.

Key components of the implementation included: - Initialization of the Q-table and exploration probability. - Definition of quantum states and measurement operators. - Execution of the Q-learning algorithm, with actions chosen based on the epsilon-greedy policy. - Update of the Q-table based on rewards received from the quantum system.

6.3 Results and Analysis

The simulation results demonstrated that the Q-learning agent could effectively learn to optimize phase shifts, achieving mean rewards close to the theoretical maximum. The mean rewards per 10 episodes for different values of N indicated that the learning process was successful, with rewards stabilizing at high values, demonstrating the effectiveness of the approach.

The analysis of the global Q-table showed that the agent learned to prefer certain actions in specific states, indicating successful learning. Additionally, the calculated value of V_H was very close to the theoretical limit, confirming the efficiency of the reinforcement learning approach.

6.4 Challenges and Mitigation Strategies

Several challenges were faced during the implementation:

- Defining an appropriate state and action space to ensure efficient learning.
- Balancing exploration and exploitation to allow sufficient exploration of the state space.

These challenges were mitigated through careful design of the state and action spaces, gradual decay of the exploration probability, and averaging results over multiple trials.

6.5 Conclusion

The adaptive phase estimation using reinforcement learning demonstrated promising results, achieving phase estimation close to the Standard Quantum Limit. The Q-learning agent successfully learned optimal strategies through interaction with the quantum system, showcasing the potential of reinforcement learning in quantum computing applications. The adaptability and efficiency of the reinforcement learning approach offer significant advantages over traditional methods, making it a viable technique for advanced quantum systems.

Future work could explore the application of other reinforcement learning algorithms, such as deep Q-networks or policy gradient methods, to further enhance the efficiency and accuracy of phase estimation in quantum systems. Additionally, extending the approach to more complex quantum states and larger systems could provide deeper insights into the potential of reinforcement learning in quantum computing.

Bibliography

[1] Kartikeya Rambhatla, Simone Evaldo D'Aurelio, Mauro Valeri, Emanuele Polino, Nicolò Spagnolo, and Fabio Sciarrino. Adaptive phase estimation through a genetic algorithm. *Phys. Rev. Res.*, 2:033078, Jul 2020.