Adaptive Phase Estimation using Reinforcement Learning

- Manish Mallapur

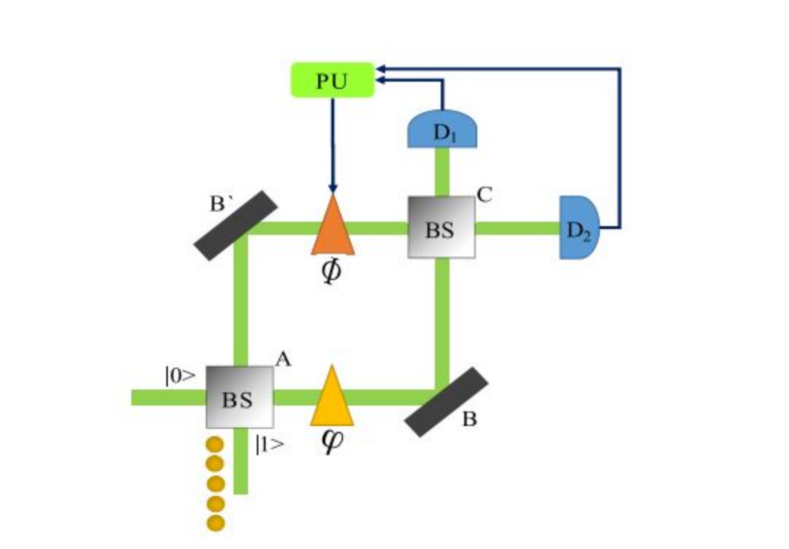
Introduction

In this project, I've used Q-learning, a type of reinforcement learning, to improve phase estimation in a Mach-Zehnder interferometer (MZI). A MZI helps estimate an unknown phase shift between two light paths by measuring the interference pattern. Traditional methods can struggle with limited data. By introducing an adaptive protocol and using Q-learning to adjust an extra phase shift based on previous measurements, we aim to make phase estimation more accurate and efficient.

Progress:

- Simulating the Mach-Zehnder interferometer
- Used reinforcement learning to optimize for the angle in the adaptive phase estimation.
- Verifying the Holevo Variance (did it for N=15 to 19, angle = Pi/2)

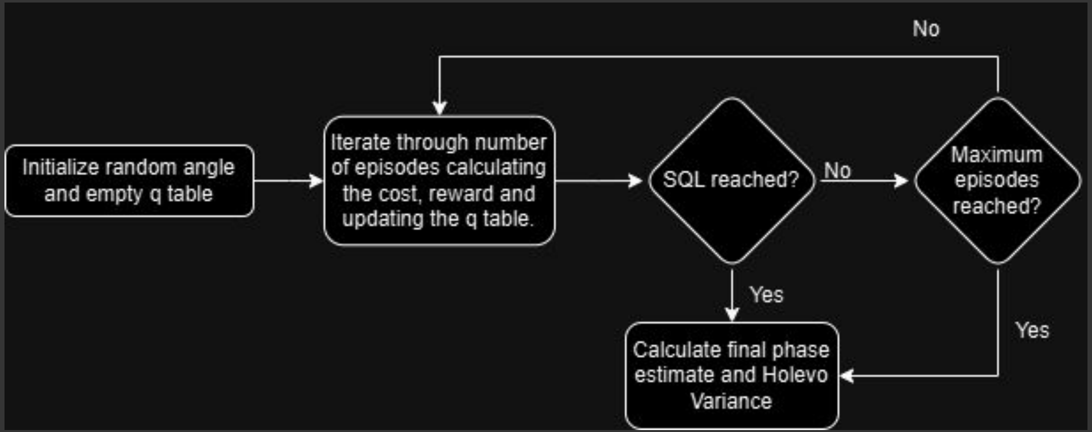
The Experimental Setup:



It Contains:

- Classical Processing unit
- 2 Photon Detectors
- 2 Beam Splitters
- 2 Mirrors
- Controllable phase

Implementation:



Phase Estimation:

- Just looking at the state with the highest q value for action 0 should work

OR

- To do this, we look at the q table and choose all the states with q value for action 0 greater than for actions -1 and 1.
- Calculate the cost for each of the states.
- The state with the lowest cost and highest q value is the required state.

Challenges:

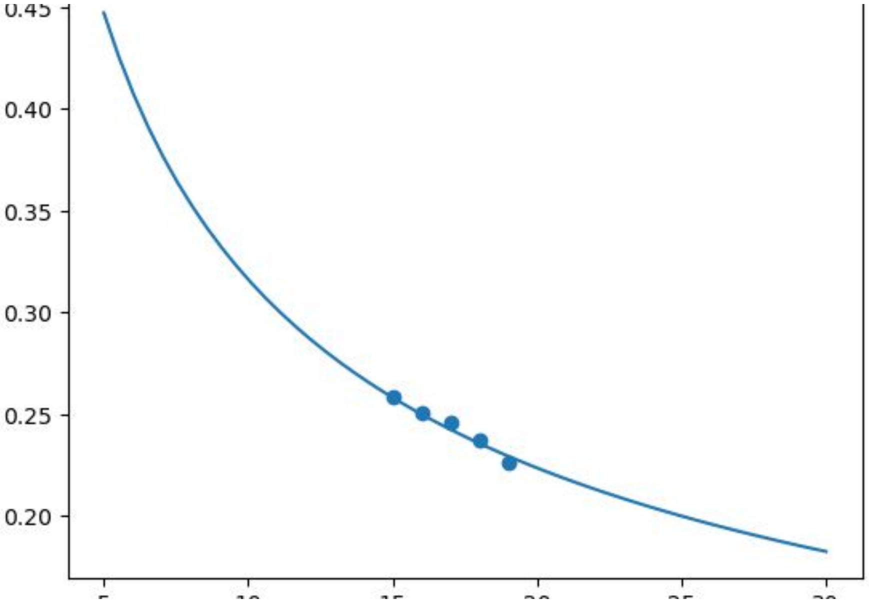
- Extremely long simulation times.
- Coming up with an efficient cost function.

The Results

```
[0.         0.         0.        ]
[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]
[1.11433732 0.22849897 0.37294381]
[0.17156155 0.13279494 0.63571518]
[0.15363347 0.07013707 0.17680676]
[0.66774848 0.16511288 0.22910313]
[0.08175196 0.         0.21543148]
[0.         0.         0.        ]
[0.         0.         0.        ]
```

The Q Table:

- For N = 16
- After 100 Episodes
- Angle: Pi/2



The Holevo Variance:

- For N = 15 to 19
- After 100 Episodes
- Angle: Pi/2

What’s Next:

- Running multiple instances of the simulation with different N on the cloud instead of locally.
- Using squeezed light to achieve better limit(Heisenberg Limit) than the Standard Quantum Limit.

Thank You!