COMPARISON STUDY OF REINFORCEMENT LEARNING ALGORITHMS

Anubhav Shrimal(MT18033), Mahika Wason(2016241), Rupal Jain(2015081)

I. INTRODUCTION

IN this project, we try solving virtual enviroment problems in Open AI gym using the reinforcement paradigm of an actor-critic. An actor performs specific actions in its environment which are rewarded by the critic. The actor learns from these feedbacks and carries out future activities with an an aim to improve upon them. We analyze and compare the performance of different reinforcement based techniques.

II. CURRENT PROGRESS

We have implemented Q learning algorithm for two environments in Open AI gym: FrozenLake-v0, Taxi-v2. The algorithms were tested by varying various parameters like discounting rate, decay rate, learning rate and episode count.

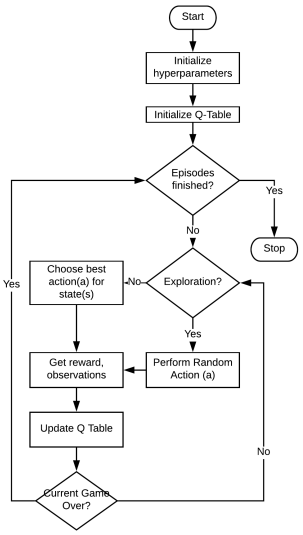
III. ALGORITHM IMPLEMENTATION(Q LEARNING)

Bellman Equation to update Q-Table:

$$NewQ(s,a) = Q(s,a) + \alpha[R(s,a) + \gamma max Q'(s',a') - Q(s,a)]$$

1. Initialize Q-values (Q(s,a)) arbitrarily for all state-action pairs.
2. For life or until learning is stopped...
3. Choose an action (a) in the current world state (s) based on current Q-value estimates (Q(s,.)).
4. Take the action (a) and observe the outcome state (s') and reward(r).
5. Update Q(s,a):= Q(s,a) + $\alpha[r + \Gamma max_{a'})Q(s',a') - Q(s,a)]$

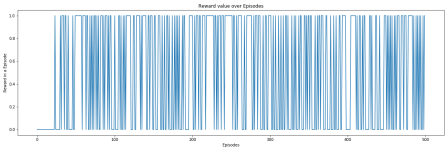
IV. BLOCK DIAGRAM



V. RESULTS

TABLE I
HYPERPARAMETERS TUNING

Number of Episodes	10000	15000	20000
Average Reward	0.472	0.48	0.4897
Learning Rate	0.8	0.9	0.6
Average Reward	0.472	0.4584	0.5072
Discount Rate(gamma)	0.95	0.75	0.99
Average Reward	0.472	0.377	0.4692
Decay Rate(epsilon)	0.005	0.008	0.003
Average Reward	0.472	0.4822	0.466



VI. REMAINING WORK

Q learning has only been able to solve simpler environments like FrozenLake and Taxi. We will be using more advanced Reinforcement Learning algorithms of Deep Q learning to solve Pacman and Atari breakout. We will then be comparing these algorithms' performance over different environments.

VII. REFERENCES

https://link.springer.com/article/10.1007/BF00992698

VIII. CONCLUSION

We conclude that due to large space requirements of Q learning algorithms, they are not feasible for complicated problem statements with large number of states. Q learning