Assignment II

Comparison of Q Learning, DoubleQ Learning, Triple Q Learning and Quadruple Q Learning

using the cliff walking problem

REINFORCEMENT LEARNING
3-1,2024

Apoorva Gayatri K
2022A7PS0232H

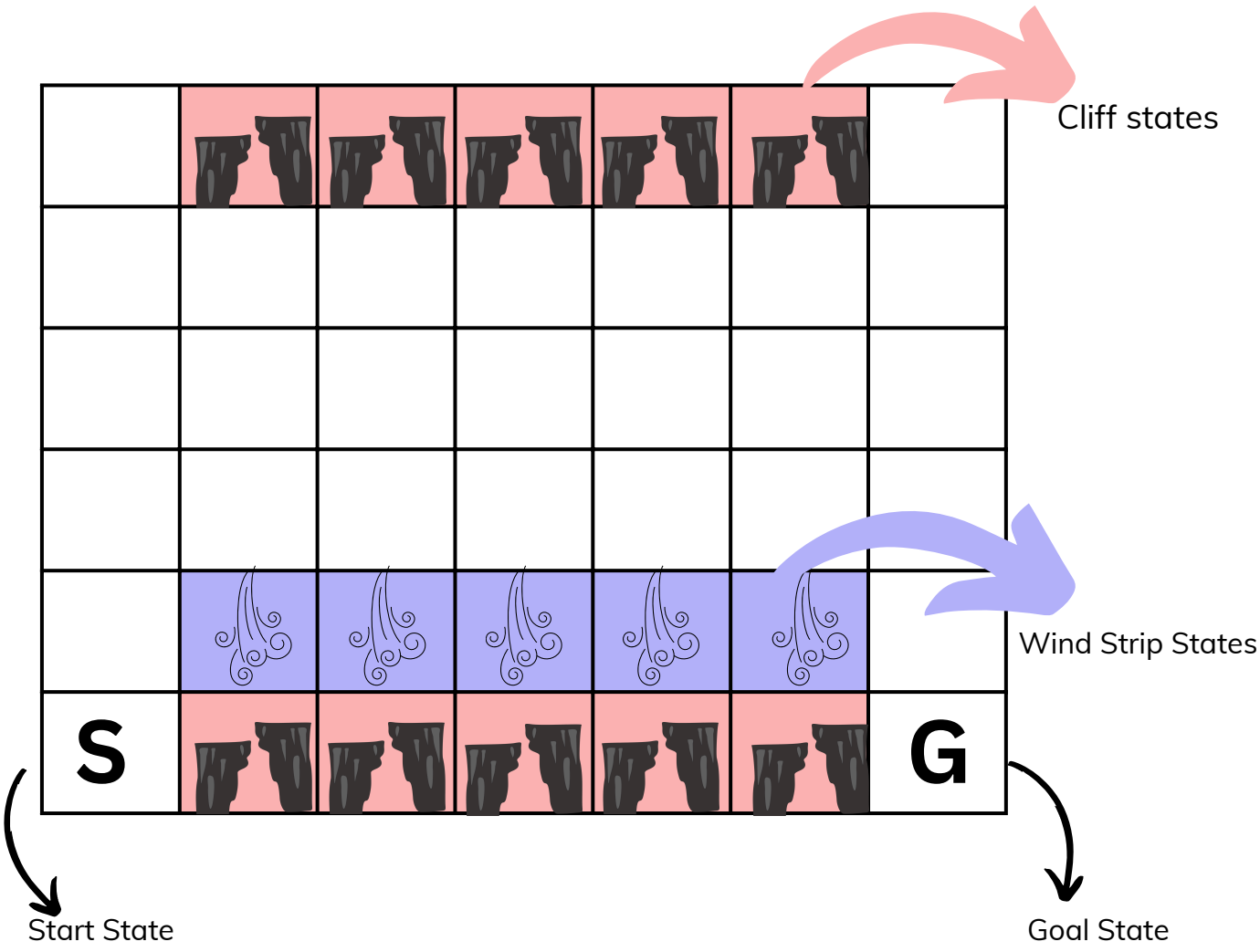
Problem Statement:

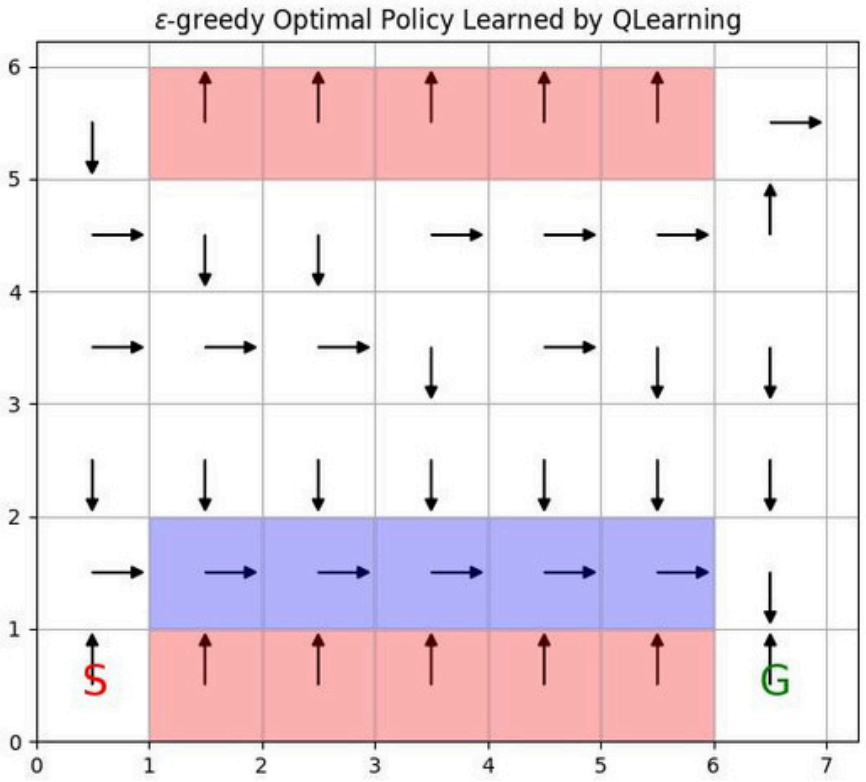
This cliff world problem is defined by a grid of size of 6 X 7 grid where there are two cliffs. One in the 0th row between the start and the end states and one near the 5th row(zero based indexing).

The stochasticity to the problem is added with *wind*, *i.e* when the agent is directly above the bottom cliff, and attempts to move right there is a 50% probability that it goes down and falls to a cliff state. When an agent reaches a cliff state then the agent obtains a significant negative reward and goes back into the start state.

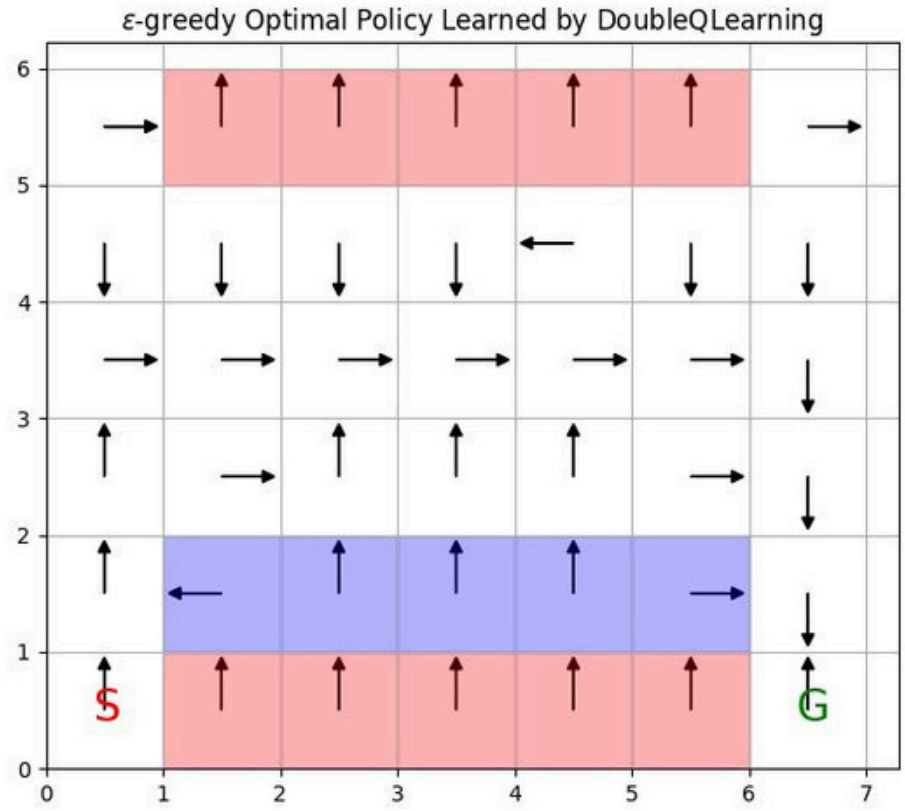
Every transition to a non terminal state (i.e the goal) results in a minimal negative reward. This is done with the intention that the model tries to learn the shortest possible path that reaches the goal state.

The model was trained with all the policy learning methods, for 5000 episodes and the average reward per episode was plotted with a batch window size of 50, to track convergence

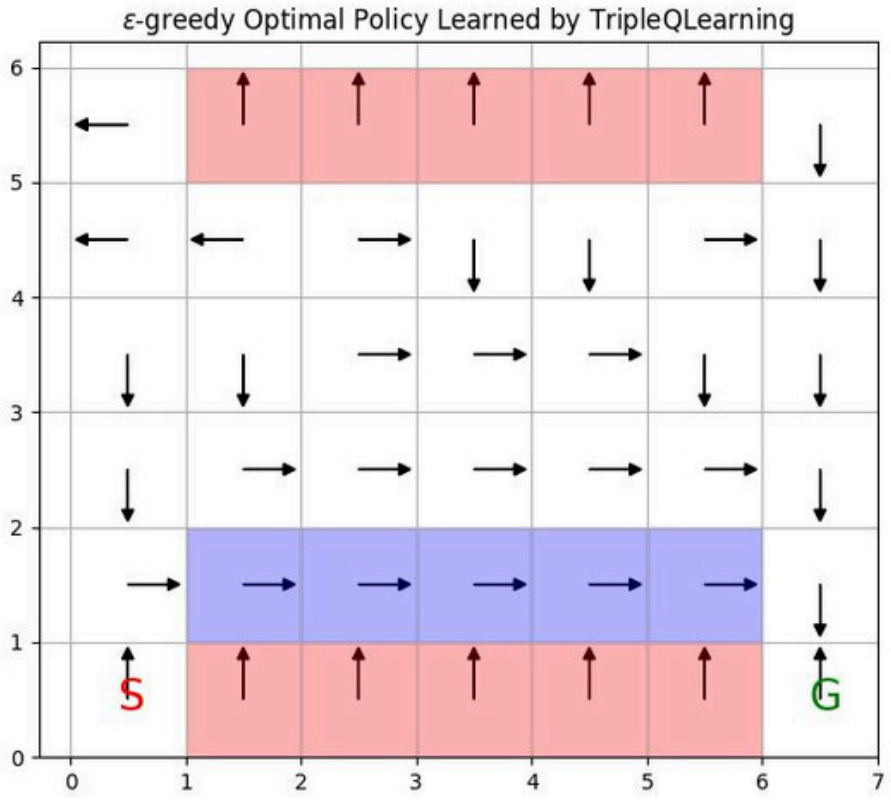




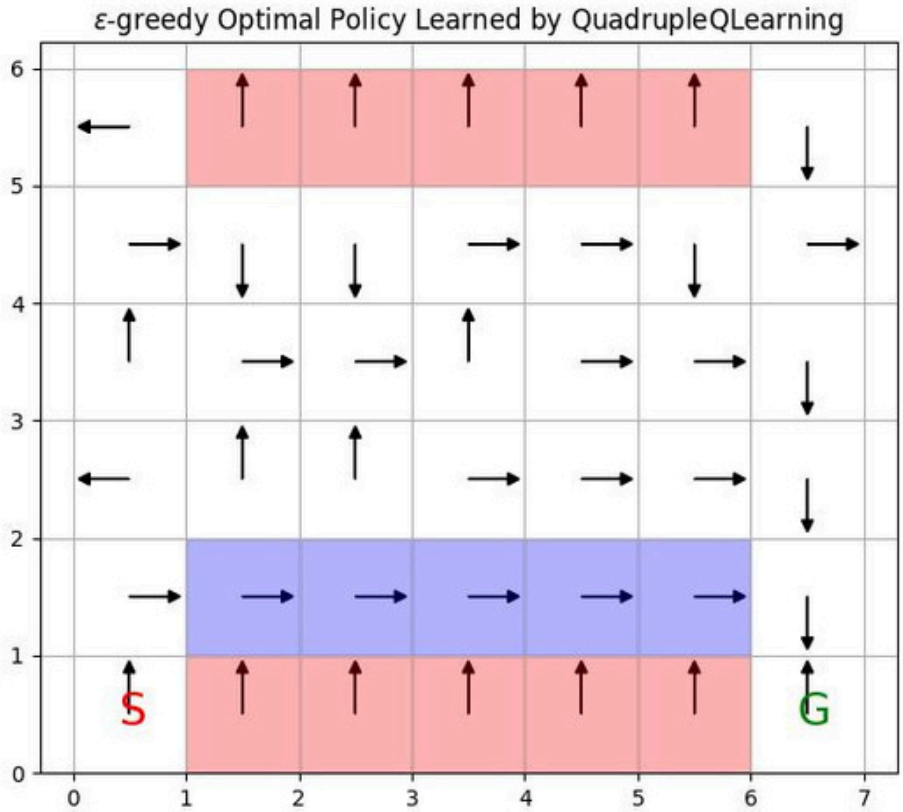
Q learning learns policy that goes through wind strip



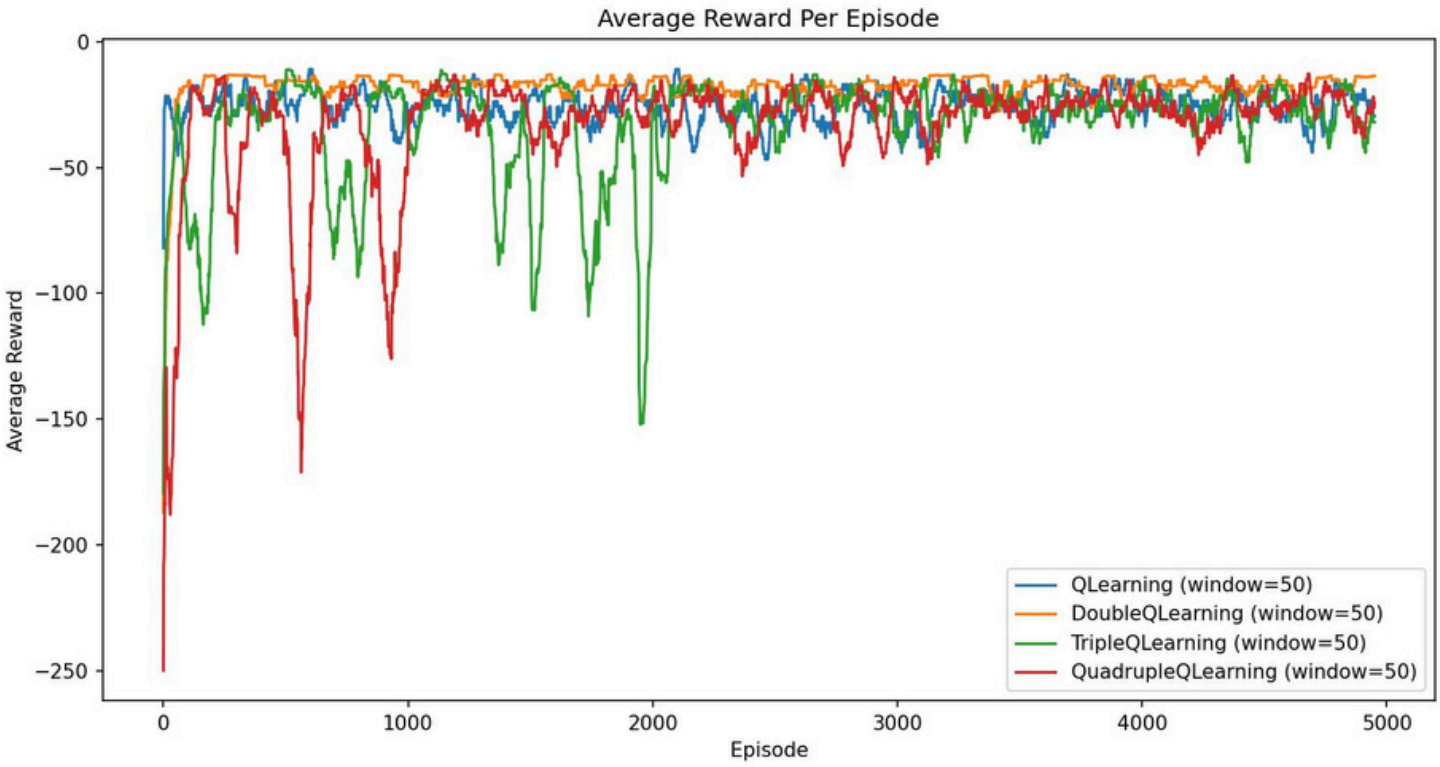
Double Q learning learns policy that goes that avoids the wind strip



Triple Q learning learns policy that goes through wind strip



Quadruple Q learning learns policy that goes through wind strip



Observations:

- We see that the average reward of Double Q learning is consistently higher than that of the Q Learning method.
- We see that Double Q learning is much more stable than Q learning
- Triple Q learning and Quadruple Q learning, despite reaching a convergence are much more unstable that Q learning itself.
- The policy learnt by Q learning,Triple Q learning and Quadruple Learning are identical and provide the same path from S to G.
- Double Q learning on the other hand provides an optimal policy avoiding the 1st and the 2nd row.
- Triple Q learning approximately gains the same average reward as that of Q learning after around 2000 episodes
- Quadruple Q learning is the most unstable and does not have consistency in its average reward however it tends to achieve a stability around 3000 episodes
- The most stable parts of the triple or quadruple Q learning average rewards are still more unstable than the Double Q learning average rewards.
- Double Q learning provides with the most stable average returns per episode out of all the three learning methods indicating reliability of the actions learnt by the policy.

Analysis:

- Double Q learning proves to be the learning method- it has the most stable average rewards per episode that are consistently higher than any of the other learning methods- namely Q learning, Triple Q learning and Quadruple Learning
- The reason behind Q learning's instability and lower average is due to the maximisation bias brought in with the stochastic nature of the strip above the cliff with the wind.
- The wind enables 50% of the attempts that the agent, while in the blue strip, tries to go right, the agent ends up going down instead.
- But as Q learning only takes in the maxarg of Q(s,a) this can lead to an overestimation bias that is never corrected and the policy chooses the same action that causes the agent to fall down from the wind strip to the cliff.
- This is provides sub-optimal policy where you end up choosing the action of max Q value but 50% of the time the action takes decreases your return as you gain a negative reward
- Double Q learning handles this overestimation, as even though Q1 learns this action to be the best action, Q2 updates its corresponding value using the stochastic action that is being overestimated by Q1. If 50% of the time the action makes you fall down instead and gain a negative reward this is updated in Q2 and Q2 will reflect a lower value than that estimated by Q1
- The policy would be argmax(E(Q)) for double Q learning this would be (Q1+Q2)/2 so even if Q1 is overestimated the other Q2 balances the overestimation.
- Triple Q learning and Quadruple Q learning on the other hand do not prove to be as effective.
- The reason Triple Q learning and Quadruple Q learning fail due to the increased randomness brought in by more than one Q tables present for updation of one Q table

```
if table_choice == 1:
    td_target = reward + self.discount * max(self.q2[state, self.argmax(self.q1[state, :])], self.q3[state, self.argmax(self.q1[state, :])])
    self.q1[self.prev_state, self.prev_action] += self.step_size * (td_target - self.q1[self.prev_state, self.prev_action])
```

```
if table_choice == 1:
    td_target = reward + self.discount * max(self.q2[state, self.argmax(self.q1[state, :])],
                                             self.q3[state, self.argmax(self.q1[state, :])],
                                             self.q4[state, self.argmax(self.q1[state, :])])
    self.q1[self.prev_state, self.prev_action] += self.step_size * (td_target - self.q1[self.prev_state, self.prev_action])
```

- When we update the Q1 value by taking the max of Q2,Q3,Q4 with (action as the maxarg of Q1) then even if Q2, Q3 or Q4 is trying to decrease the overestimation of Q1 it fails to do so as the final value taken is the max state action values among Q2,Q3,Q4. Which defeats the purpose of reducing overstimation by Q learning

- This explains why triple Q learning and Quadruple Q learning tend to have the same policy as Q learning, they never correct the overestimation.
- The increased time for convergence compared to general Q learning despite converging to the same policy can be due to the fact that triple Q learning and quadruple Q learning have 3 and 4 Q tables to evaluate which naturally makes the method time intesive.

Conclusion:

- Double Q learning provides the best policy among all the four learning methods for stochastic environments
- Having more Q tables doesn't necessarily mean optimal or faster convergence
- If increasing Q tables could provide better results we would not stop at Double Q learning
- Even if we do not choose the max Q(s,a) to update and choose a value randomly, the increased uncertainty might lead to a policy that is worse than the one learnt by Q learning as the policy would be no better than a random policy.
- Q learning successfully learns the true Q values of the wind strip states and the actions associated with reaching the states.
- Optimal policy is avoiding the wind strip even if it is the shortest path as 50% of the time you take an action to the right it ends up falling from the cliff and you have to start right back from the start.