CS6700 - REINFORCEMENT LEARNING
PROGRAMMING ASSIGNMENT 1

Team member 1: Gokul Venkatesan (Roll number: ME18B047)
Team member 2: Vishal (Roll number: CH18B013)

ENVIRONMENT DESCRIPTION:

This is a grid world with 4 deterministic actions (‘up’, ‘down’, ‘left’, ‘right’). The agent transitions to the next state determined by the direction of the action chosen with a probability of ***p*** **∈ *[0, 1]***. We also define a parameter called ***b*** **∈ *[0, 1]***. Consider the direction of the action chosen as the agent’s “North”. For example, if the action is ‘left’, it is the agent’s North, and the agent’s East would be the direction of the action ‘up’. Figure 1 provides an illustration of the same. The agent transitions to the state West of the chosen action with probability ***b(1−p)***, and to the East of the chosen action with probability ***(1−p)(1−b)***. The environment may also have a wind blowing that can push the agent one additional cell to the right after transitioning to the new state with a probability of 0.4. An episode is terminated either when a goal is reached or when the timesteps exceed 100. Transitions that take you off the grid will not result in any change in state. The dimensions of the grid are 10 × 10.
Rewards: -1 for normal states, -100 for restart states, -6 for bad states, +10 for goal states.

SARSA:

SARSA was implemented for this version of the grid world. For each algorithm, experiments were run with ***wind = False*** and ***wind = True***; two different start states ***(0, 4), (3, 6)***; two values of ***p (1.0, 0.7)***; and two types of exploration 2 strategies (ϵ-greedy and softmax), making it **16 different configurations in total.** For each of the 16 configurations, the best set of hyperparameters **(ϵ in ϵ-greedy exploration, temperature β in softmax exploration, learning rate α, and discount factor γ)** were determined.

Configurations	**Wind**	**Start State**	**Value of *p***	**Exploration**
1	False	(3, 6)	0.7	softmax
2	False	(3, 6)	0.7	ϵ-greedy
3	False	(3, 6)	1.0	softmax
4	False	(3, 6)	1.0	ϵ-greedy
5	False	(0, 4)	0.7	softmax
6	False	(0, 4)	0.7	ϵ-greedy
7	False	(0, 4)	1.0	softmax
8	False	(0, 4)	1.0	ϵ-greedy
9	True	(3, 6)	0.7	softmax
10	True	(3, 6)	0.7	ϵ-greedy
11	True	(3, 6)	1.0	softmax

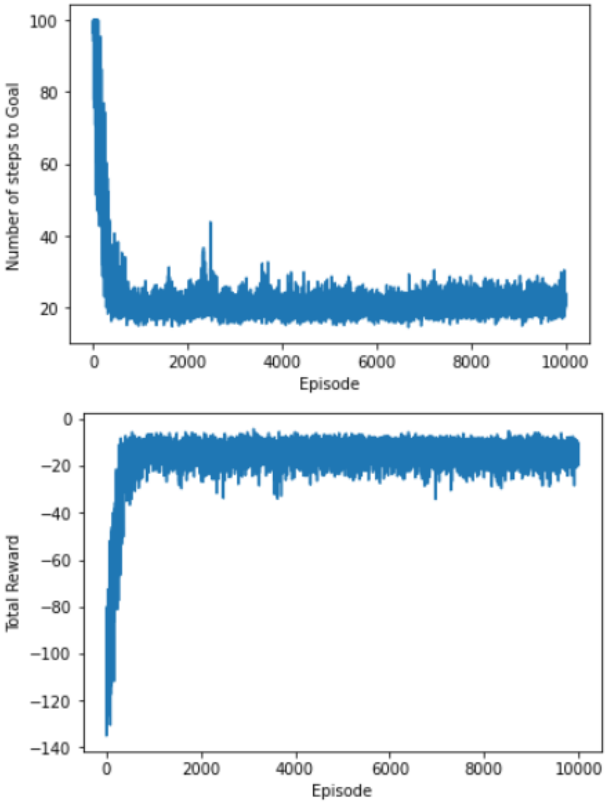
12	True	(3, 6)	1.0	ϵ-greedy
13	True	(0, 4)	0.7	softmax
14	True	(0, 4)	0.7	ϵ-greedy
15	True	(0, 4)	1.0	softmax
16	True	(0, 4)	1.0	ϵ-greedy

Table 1: Different configurations with various settings

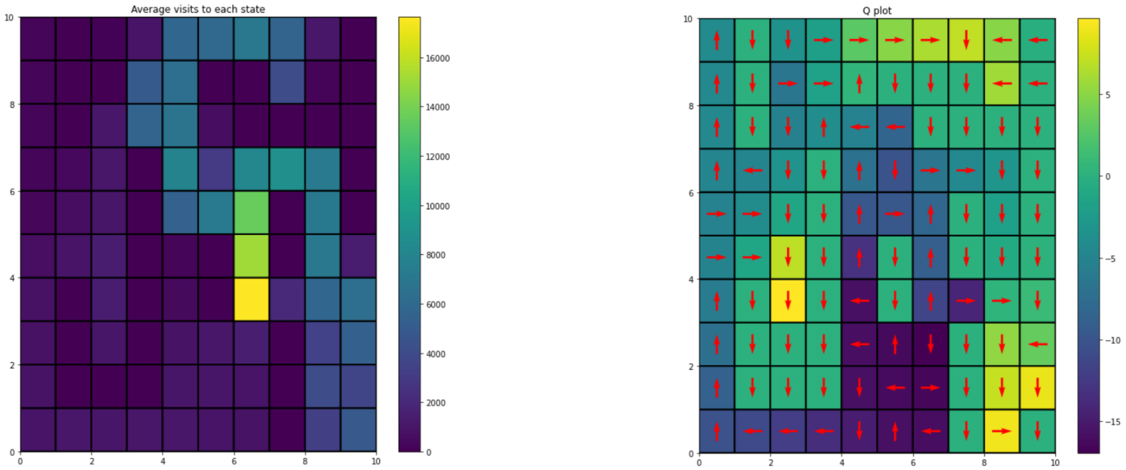
Configurations	**α**	**γ**	**β**	**ϵ**
1	0.1	0.99	0.05	NA
2	0.1	0.99	NA	0.05
3	0.1	0.99	0.05	NA
4	0.1	0.99	NA	0.05
5	0.1	0.99	0.05	NA
6	0.1	0.99	NA	0.05
7	0.1	0.99	0.05	NA
8	0.1	0.99	NA	0.05
9	0.1	0.95	0.15	NA
10	0.1	0.95	NA	0.01
11	0.1	0.95	0.15	NA
12	0.1	0.95	NA	0.01
13	0.1	0.95	0.15	NA
14	0.1	0.95	NA	0.01
15	0.1	0.95	0.15	NA
16	0.1	0.95	NA	0.01

Table 2: The set of hyperparameters for different configurations

CONFIGURATION 1:



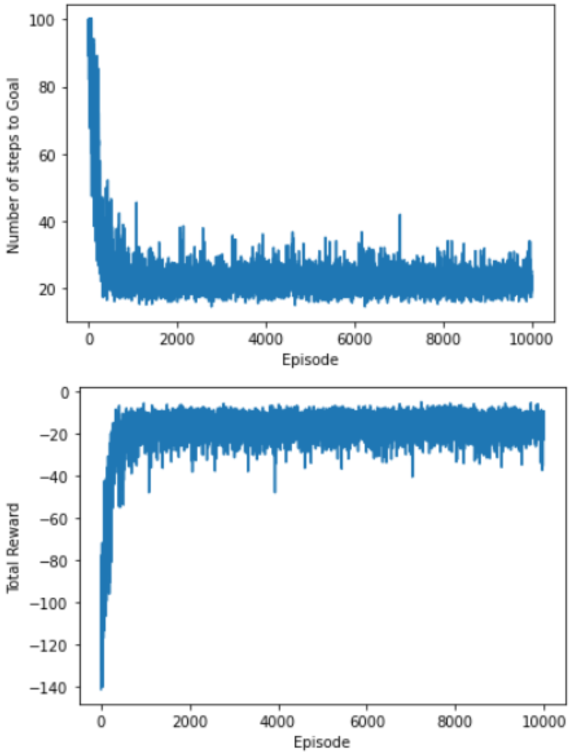
Reward curves and the number of steps to reach the goal in each episode



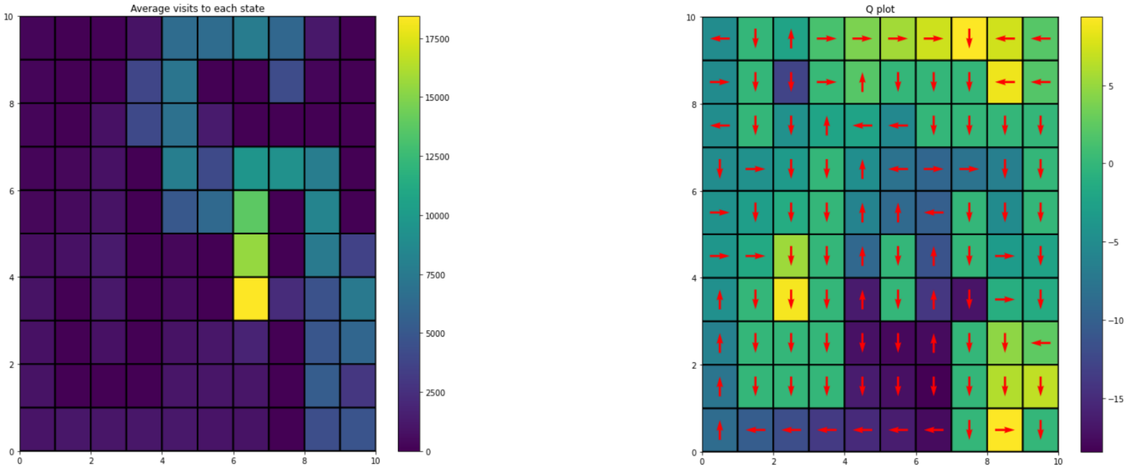
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (0, 9) or (8, 7). It does not consider the possibility of reaching the goal state (2, 2).

CONFIGURATION 2:



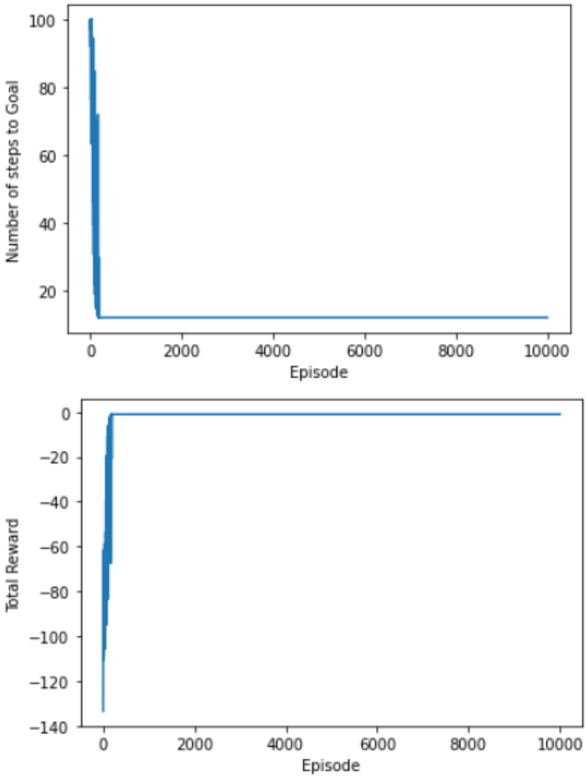
Reward curves and the number of steps to reach the goal in each episode



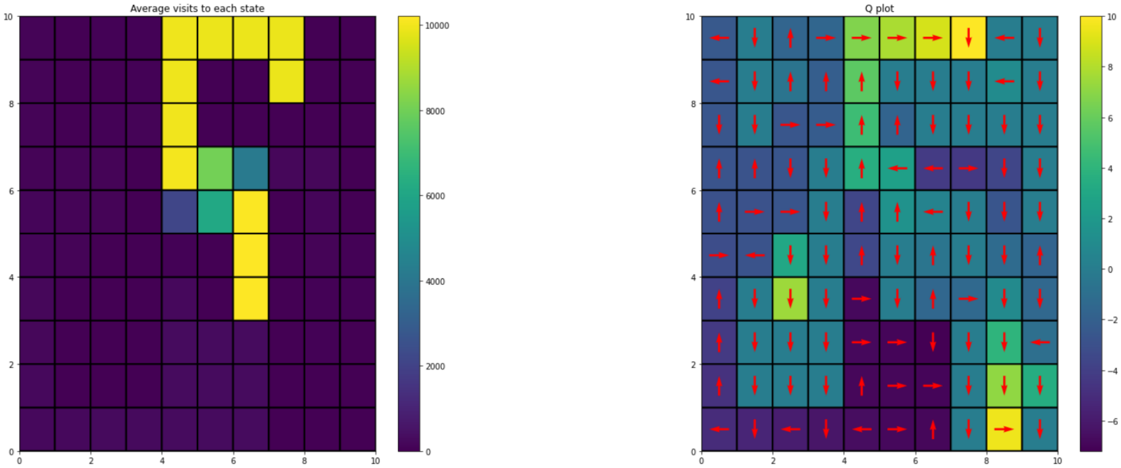
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (0, 9) or (8, 7). It does not consider the possibility of reaching the goal state (2, 2).

CONFIGURATION 3:



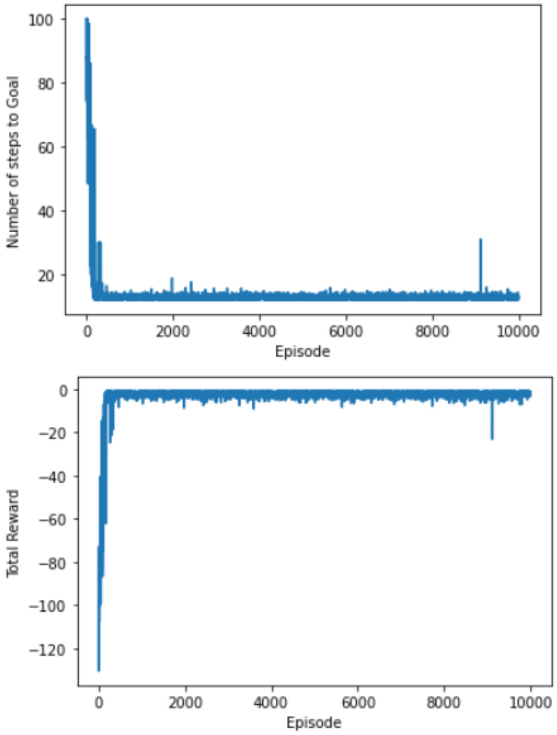
Reward curves and the number of steps to reach the goal in each episode



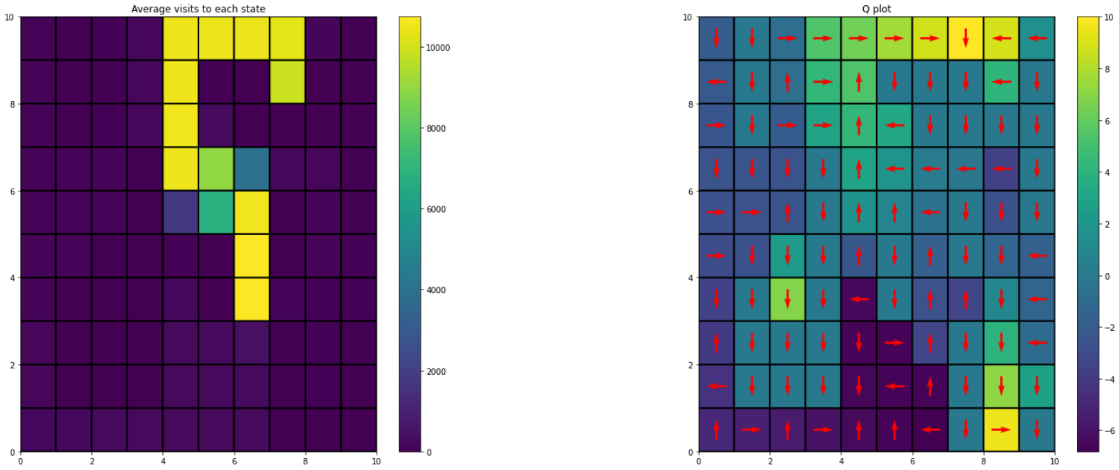
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (8, 7). It does not consider the possibility of reaching the goal states (2, 2) and (0, 9).

CONFIGURATION 4:



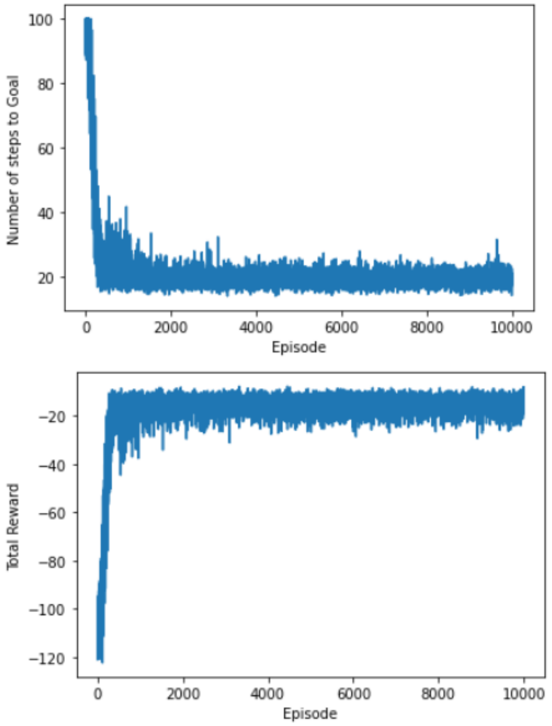
Reward curves and the number of steps to reach the goal in each episode



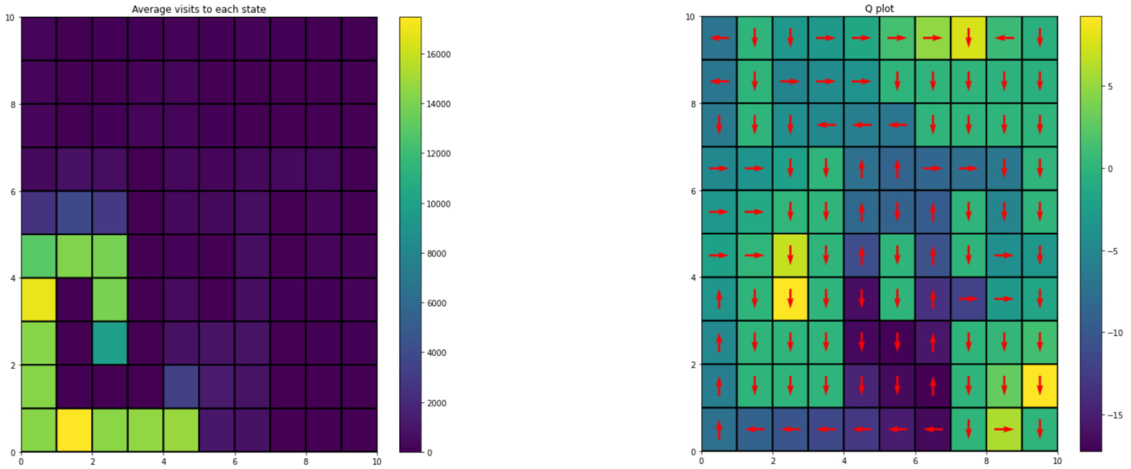
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (8, 7). It does not consider the possibility of reaching the goal states (2, 2) and (0, 9).

CONFIGURATION 5:



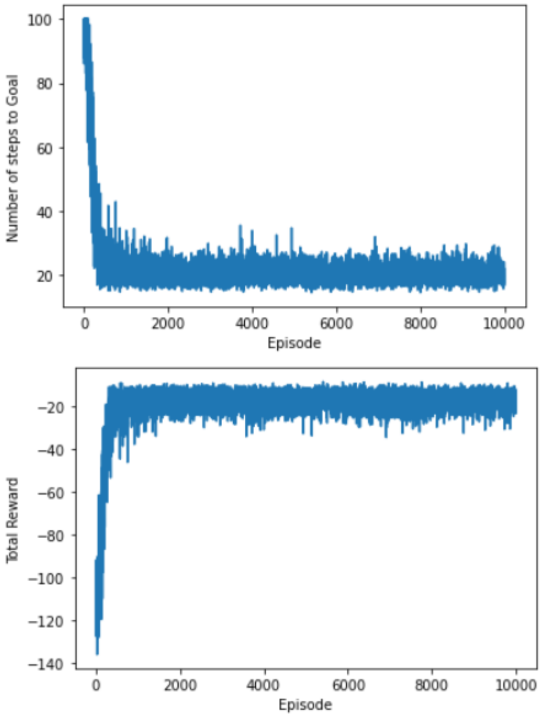
Reward curves and the number of steps to reach the goal in each episode



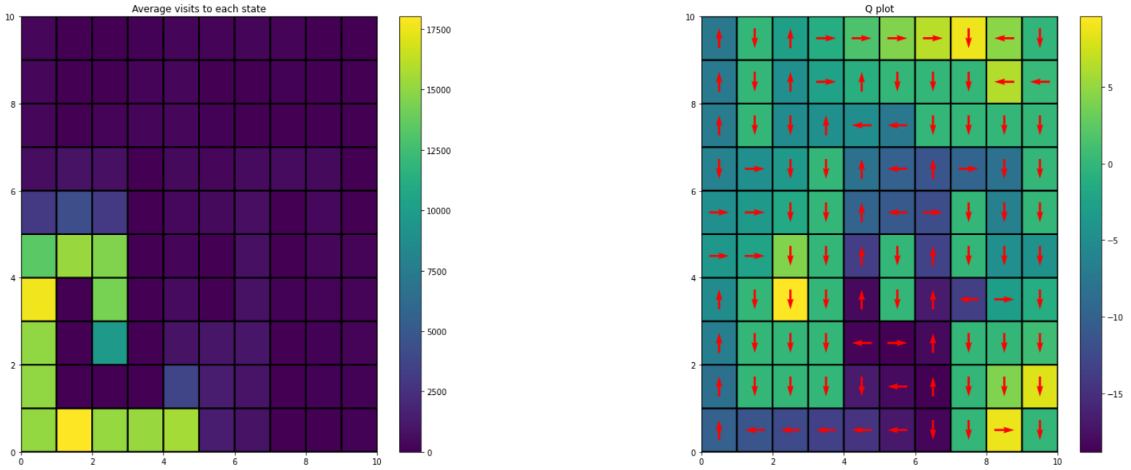
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (8, 7) and (0, 9).

CONFIGURATION 6:



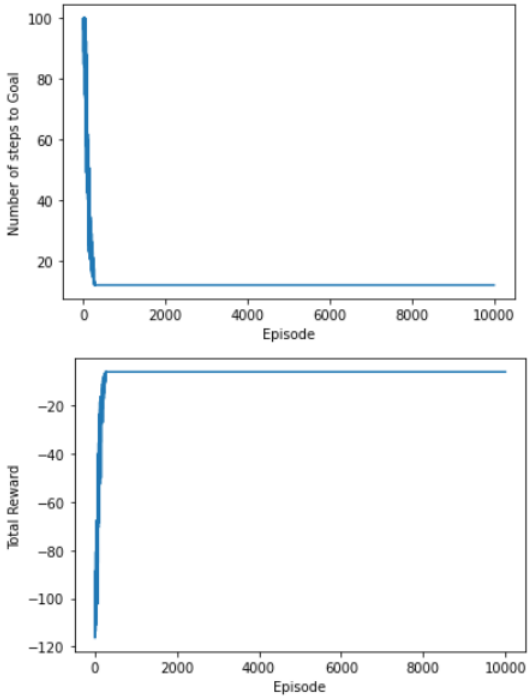
Reward curves and the number of steps to reach the goal in each episode



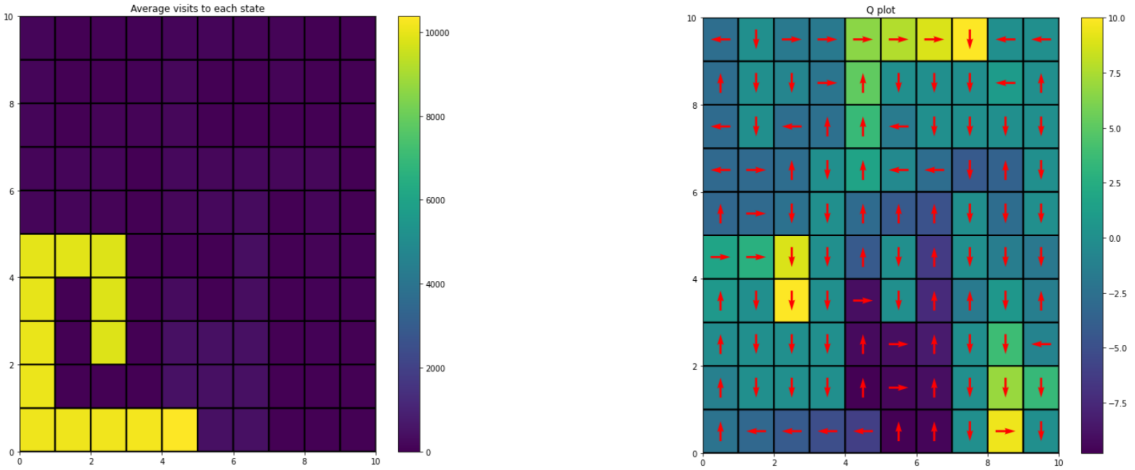
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (8, 7) and (0, 9).

CONFIGURATION 7:



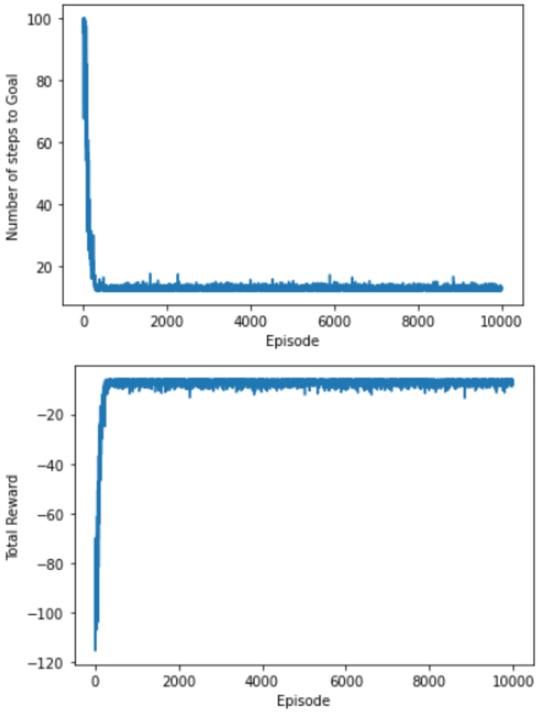
Reward curves and the number of steps to reach the goal in each episode



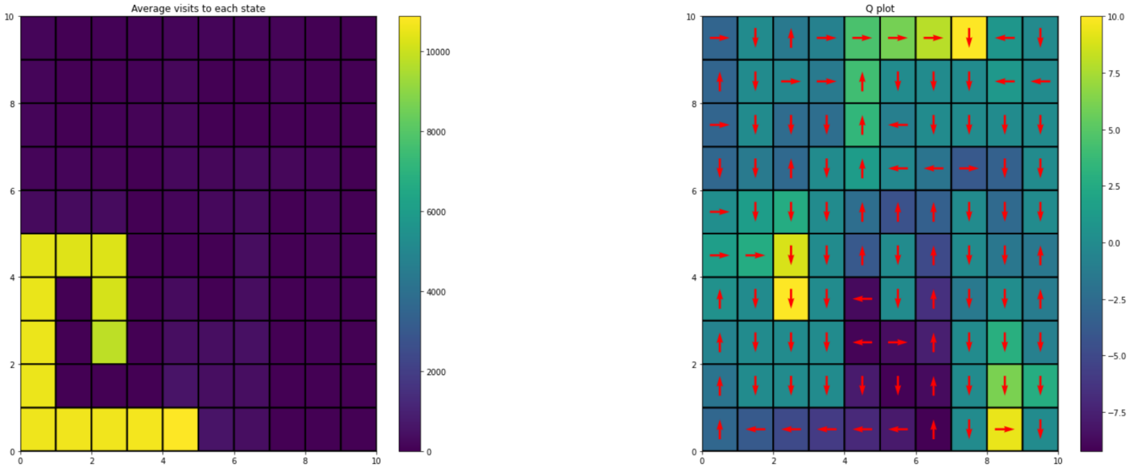
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (8, 7) and (0, 9).

CONFIGURATION 8:



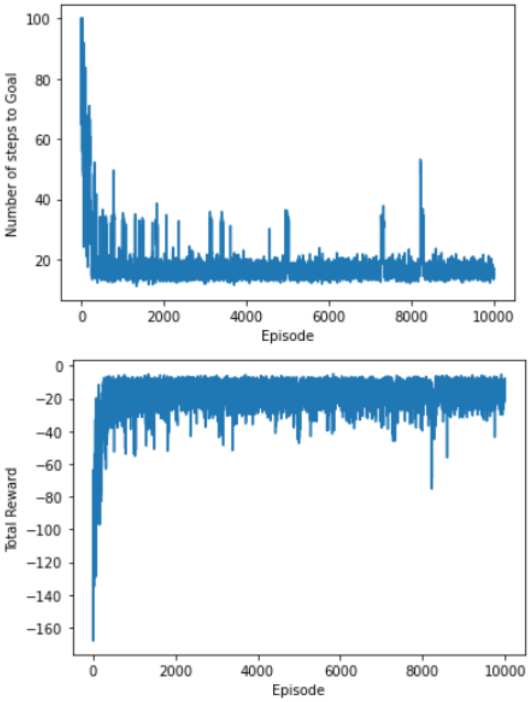
Reward curves and the number of steps to reach the goal in each episode



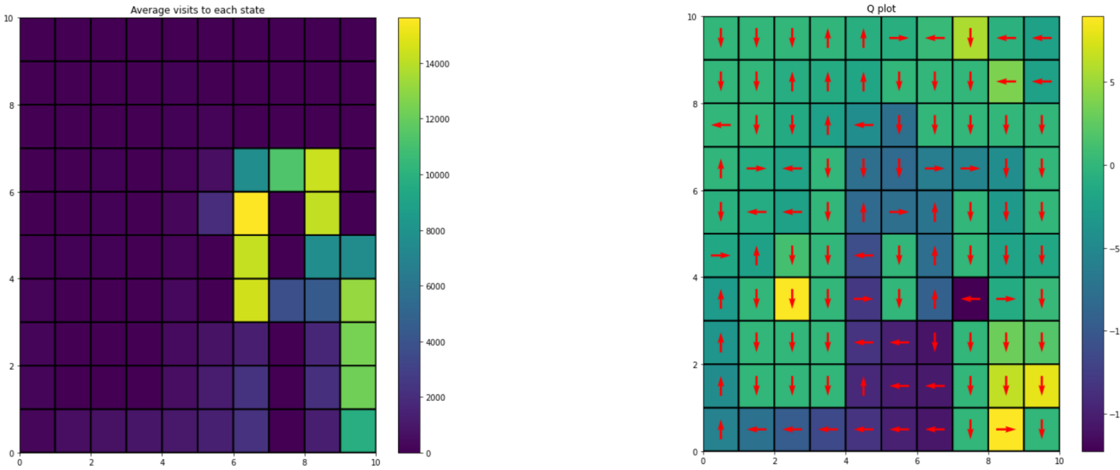
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (8, 7) and (0, 9).

CONFIGURATION 9:



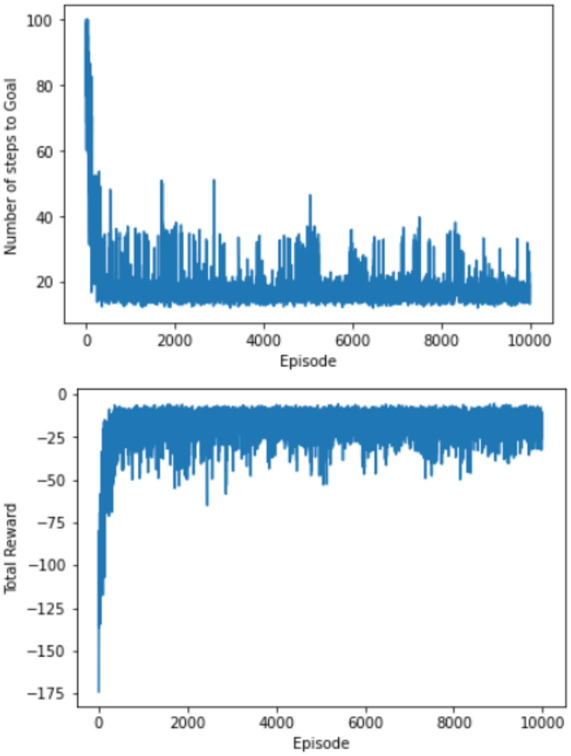
Reward curves and the number of steps to reach the goal in each episode



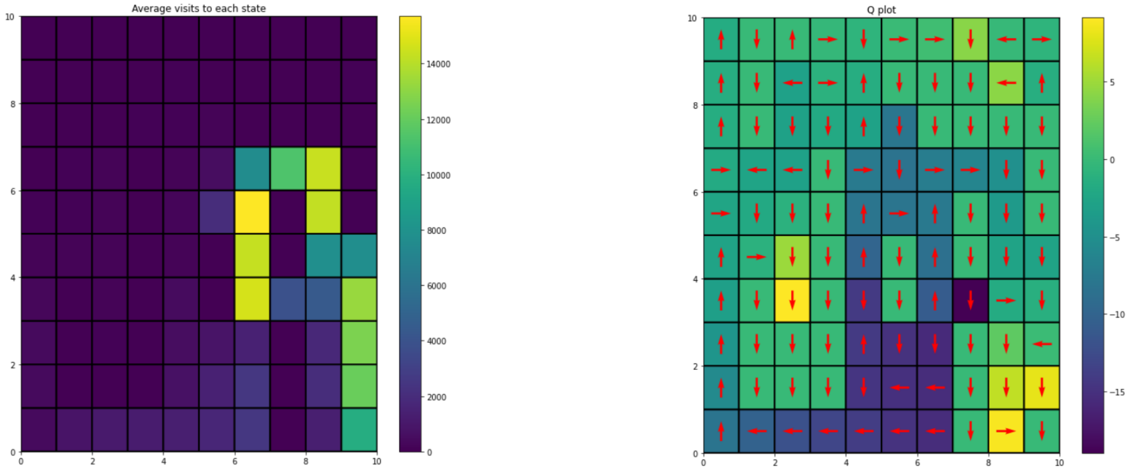
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (0, 9). It does not consider the possibility of reaching the goal states (2, 2) and (8, 7).

CONFIGURATION 10:



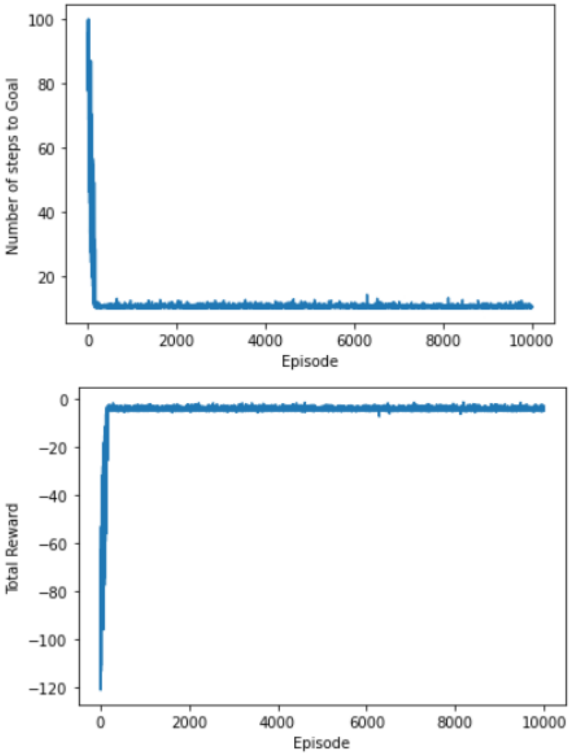
Reward curves and the number of steps to reach the goal in each episode



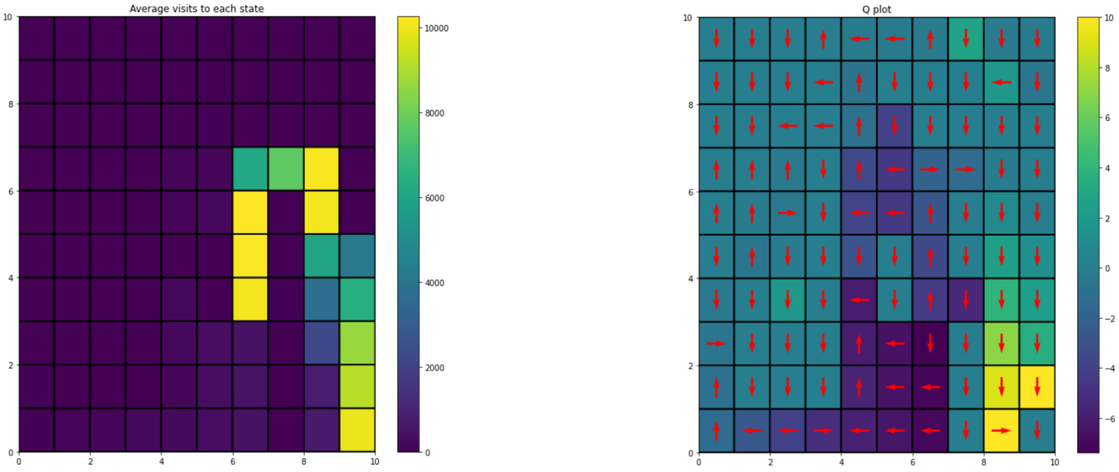
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (0, 9). It does not consider the possibility of reaching the goal states (2, 2) and (8, 7).

CONFIGURATION 11:



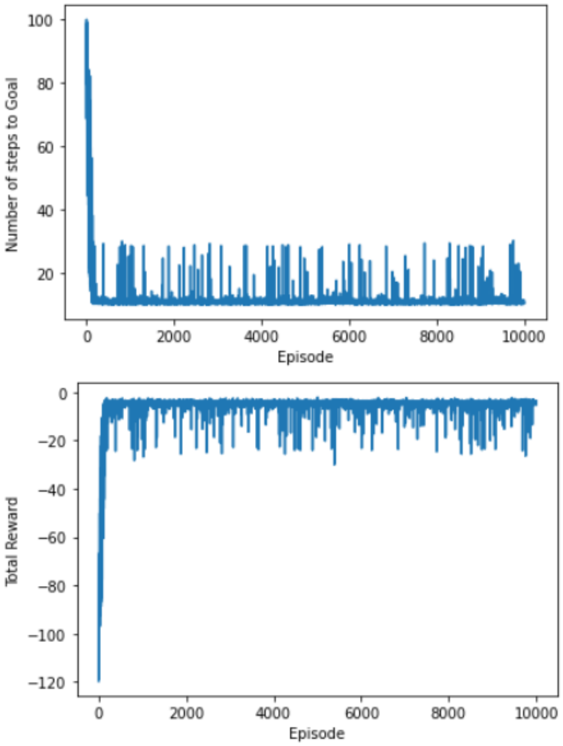
Reward curves and the number of steps to reach the goal in each episode



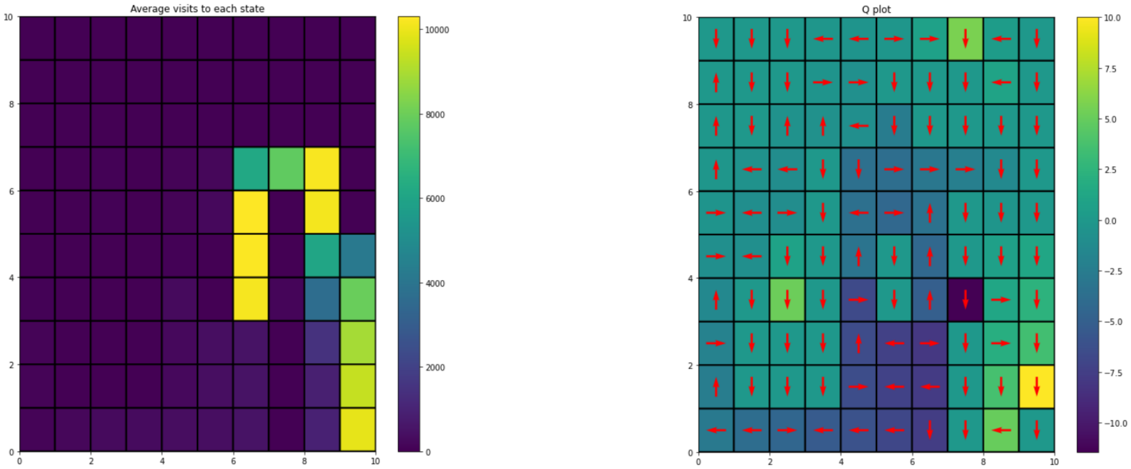
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (0, 9). It does not consider the possibility of reaching the goal states (2, 2) and (8, 7).

CONFIGURATION 12:



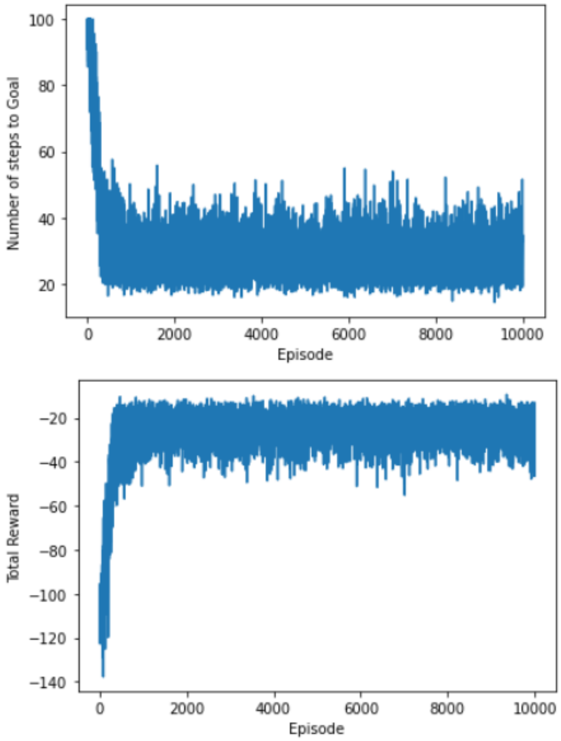
Reward curves and the number of steps to reach the goal in each episode



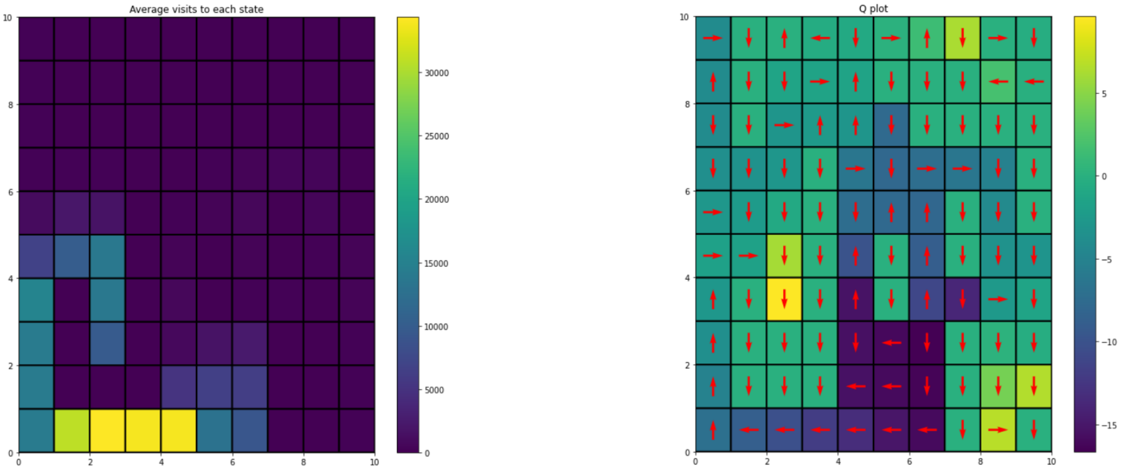
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (3, 6) and move towards the goal state at (0, 9). It does not consider the possibility of reaching the goal states (2, 2) and (8, 7).

CONFIGURATION 13:



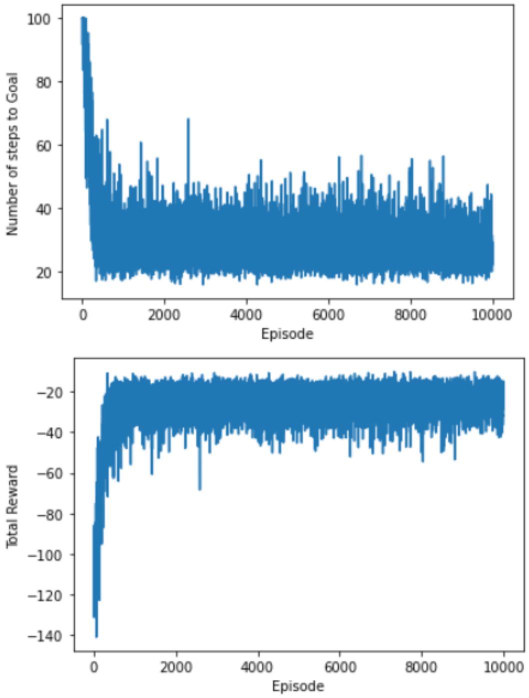
Reward curves and the number of steps to reach the goal in each episode



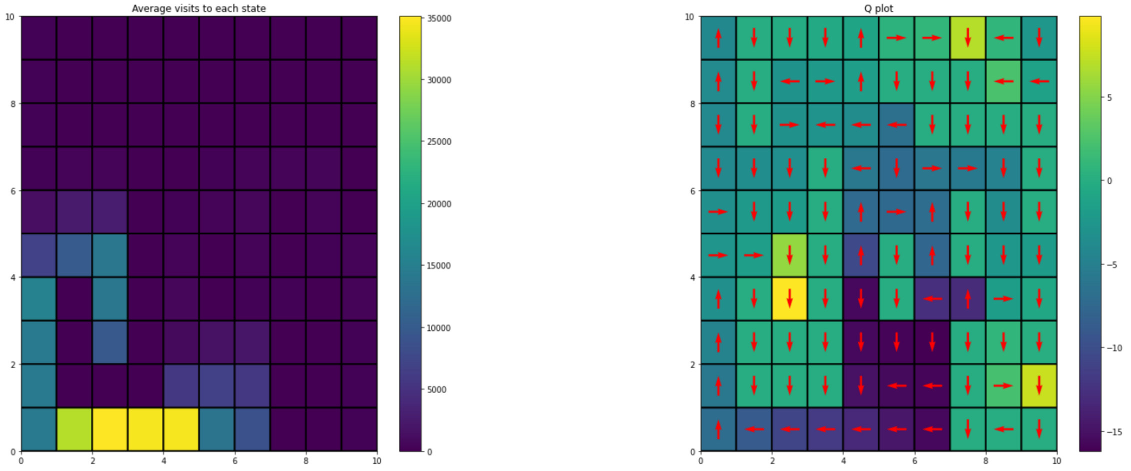
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (0, 9) and (8, 7).

CONFIGURATION 14:



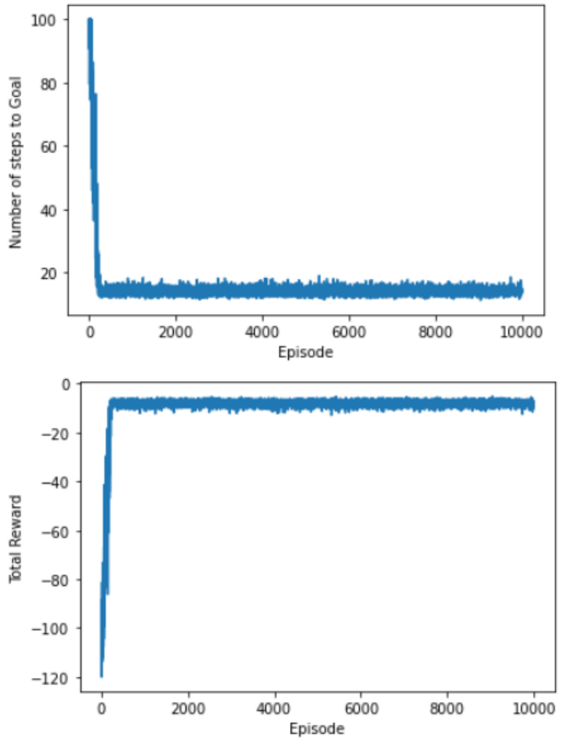
Reward curves and the number of steps to reach the goal in each episode



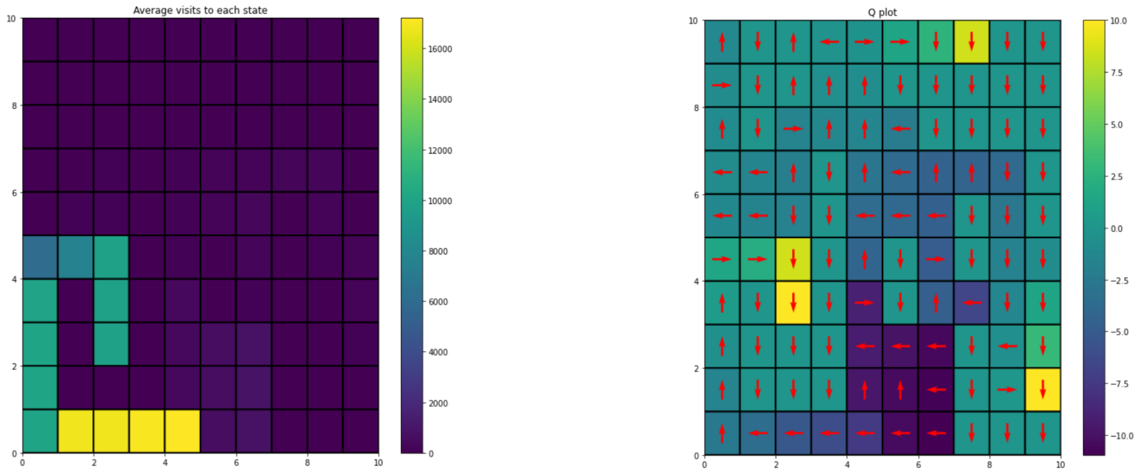
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (0, 9) and (8, 7).

CONFIGURATION 15:



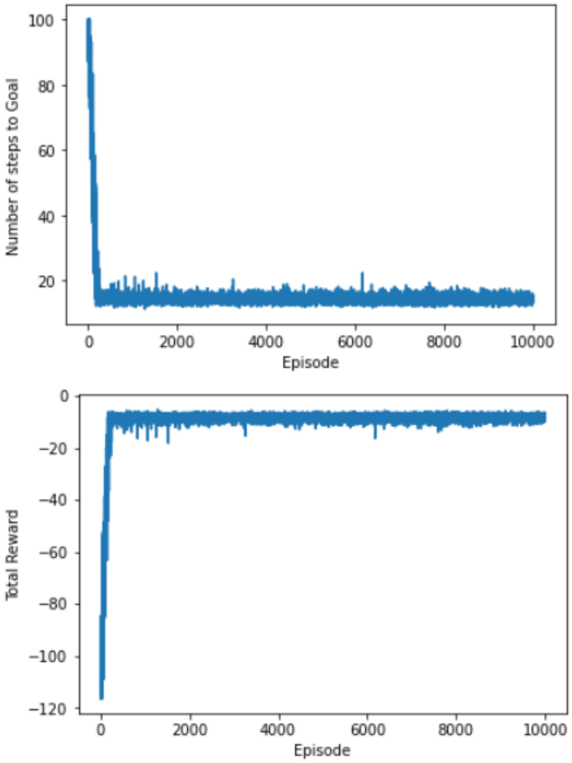
Reward curves and the number of steps to reach the goal in each episode



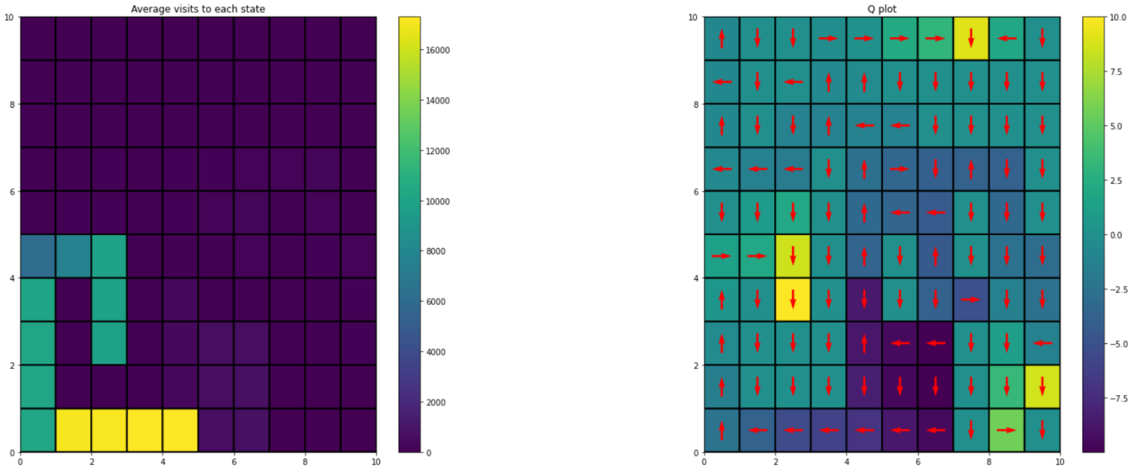
Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (0, 9) and (8, 7).

CONFIGURATION 16:



Reward curves and the number of steps to reach the goal in each episode



Heatmap of the grid with state visit counts (left) and with Q values and optimal actions (right)

For this configuration, the optimal policy learned by the agent is to start from (0, 4) and move towards the goal state at (2, 2). It does not consider the possibility of reaching the goal states (0, 9) and (8, 7).

INFERENCE:

- When the start state is (3, 6), the optimal policy learned by the agent leads to the goal states (8, 7) or (0, 9). On the other hand, when the start state is (0, 4), the optimal policy learned by the agent leads to the goal state (2, 2).
- When the probability of a good transition, ***p**,* is 1.0, the variance shown by the reward curve is very less. On the other hand, when ***p*** is 0.7, the reward curves exhibit

fluctuations. This is because of the increased stochasticity in the environment when ***p*** is 0.7.

- When ***wind = True***, the reward curves exhibit fluctuations. This is because of the increased stochasticity in the environment. On the other hand, when ***wind = False***, the variance shown by the reward curve is less compared to the case of ***wind = True***.
- When wind = False and the start state is (3, 6), the optimal policy learned by the agent leads to the goal states (8, 7) or (0, 9). On the other hand, when wind = True and the start state is (3, 6), the optimal policy learned by the agent leads to the goal state (0, 9).