Assignment-4 Report

Harsh Siloiya - I6005001I

The submission folder contains 4 files -

1. run.sh which is the main shell script used to run the program. The script takes 1 argument, which can be either ‘default’, ‘kings’ or ‘stochastic’.
2. init.py which is the entry point of the main program.
3. windygridworld.py which contains the definitions for classes for each type of windy gridworld (i.e. default, with king’s moves, and with stochasticity).
4. sarsa.py which features the implementation of the SARSA algorithm.

The values for α and ε are kept as 0.5 and 0.1 respectively. The number of episodes is kept 200, and the maximum step limit is set to be 10000. A ε-greedy algorithm is used to select an action from a state to go to the next state. This also ensures that the SARSA implementation converges.

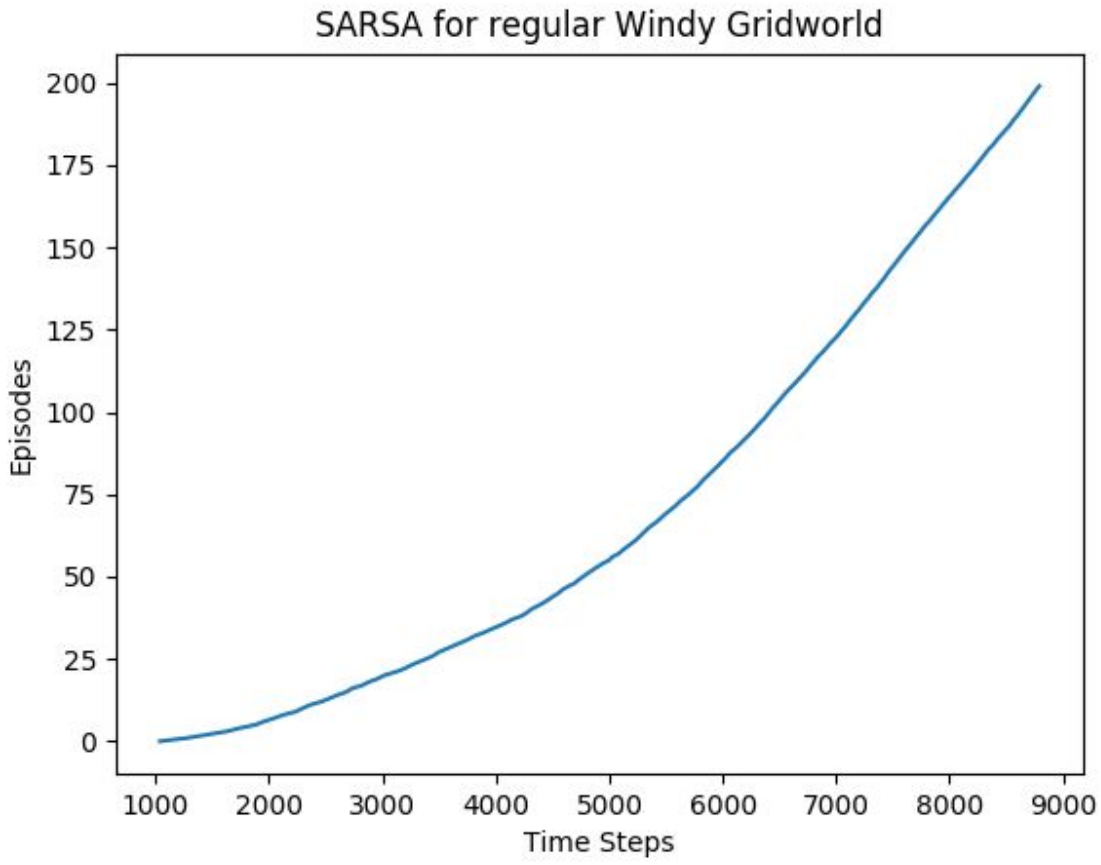
If the agent is at the border of the gridworld, moving in such a way that it exceeds border limits will force the agent to stay at its original place. This will happen for all gridworld instances (default, kings, stochastic).

In the case of stochastic winds, in the presence of non-zero wind, the agent moves to the intended position with probability ⅓, to the position one above the intended position with probability ⅓ and to the position two places above the intended position with probability ⅓.

The graphs below are taken as the average over different seed values.

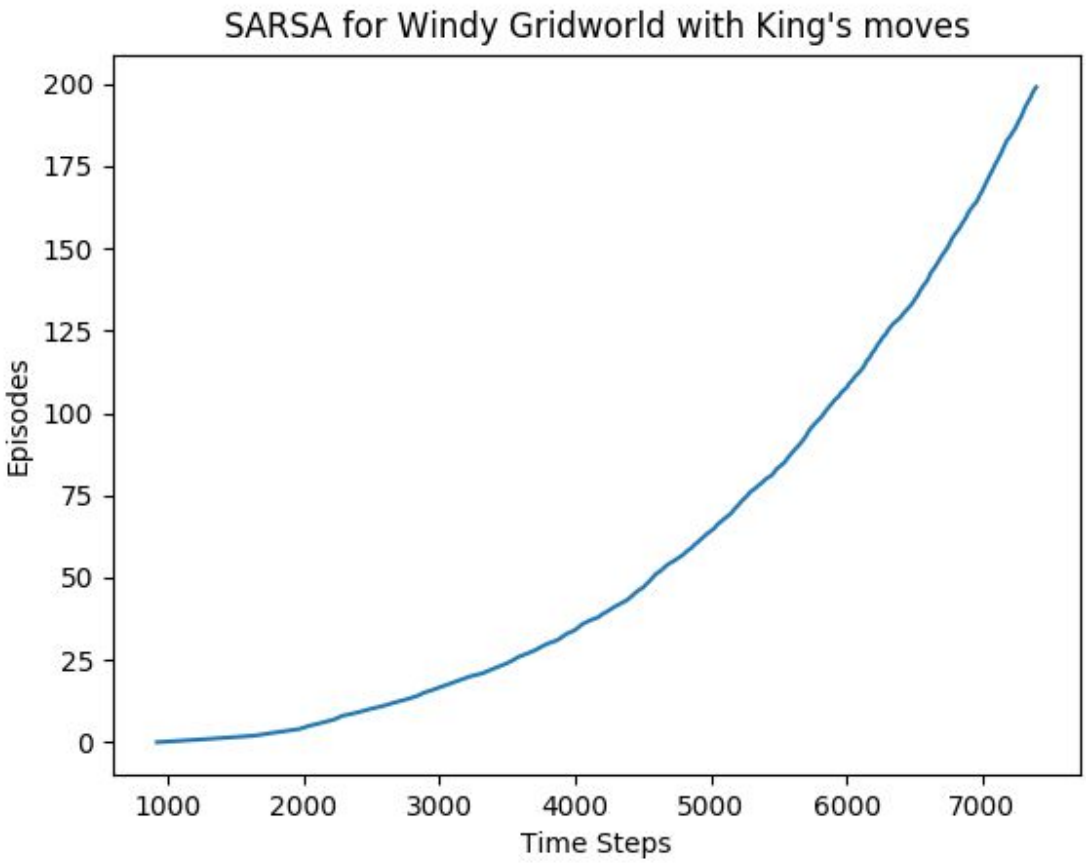
Graphs:-

For regular Windy Gridworld



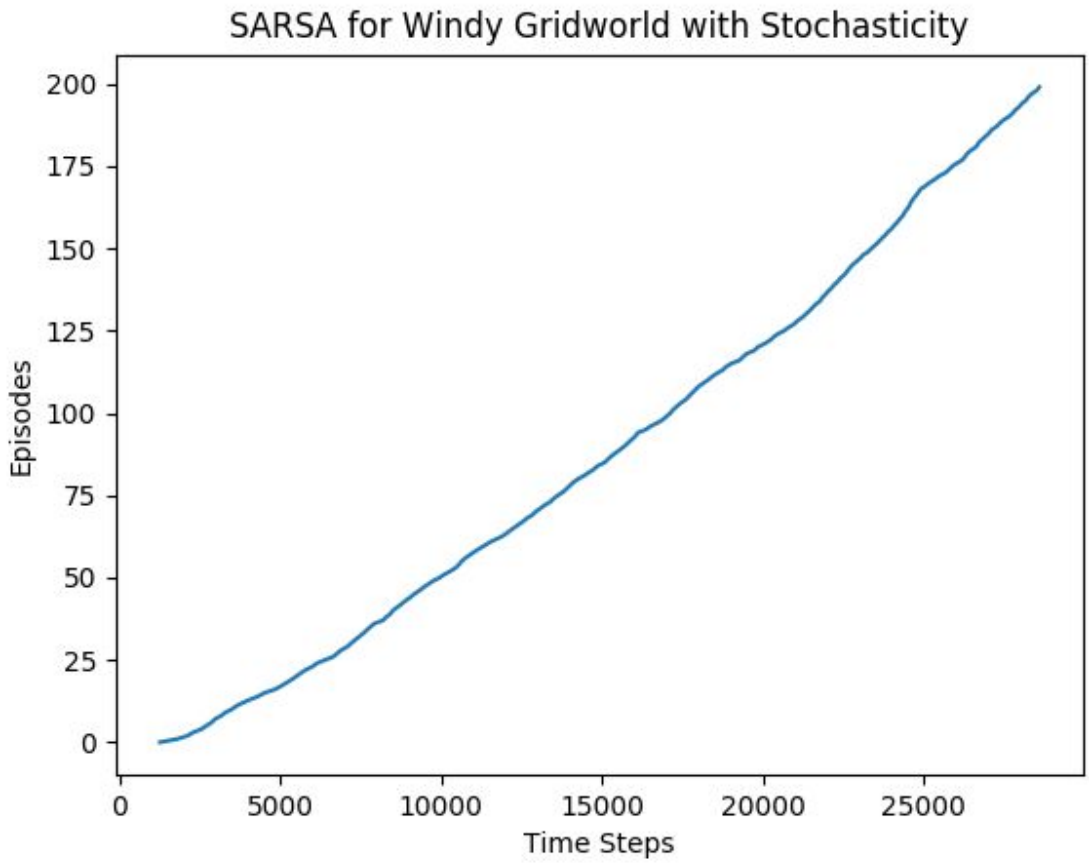
We notice that as the number of episodes increased, the slope of the graph gets steeper. This shows that the agent reaches the goal rather quickly after sufficient time steps when Q becomes approximately similar to Q*. This demonstrates that after some time steps, convergence has been achieved.

For Windy Gridworld with King’s moves



We notice that the graph looks similar to that of the gridworld with 4 moves. But in this case, the agent reaches the goal in lesser number of time steps than in the previous case. This shows that if the agent is allowed more flexibility, it can reach the goal faster than with a limited number of moves (faster convergence).

For Windy Gridworld with King's moves and stochastic wind



We notice that when winds are kept stochastic, it made reaching the goal more difficult for the agent since the average number of time steps for reaching the goal has increased and hence, it takes longer for the policy to converge. In this graph, it is visible that the policy has not converged yet even after 200 episodes since the number of time steps are still rising linearly instead of exponential rise which was observed in previous graphs.