



# CS 747 Programming Assignment 4

## Windy Gridworld task: SARSA (0)

Submitted By

**Krishna Wadhwani**

**160010031**

# Contents

1	Objective	2
2	Exploration and learning rates	2
3	Plots	2
4	Conclusions	4

# 1 Objective

The objective of this assignment was to solve the Windy Gridworld environment using SARSA (0) agent. Variants of Windy GridWorld such as allowance of king's moves and stochasticity have been implemented.

As given in the book, I have solved for the environment with 7 rows and 10 columns. Initial state, Terminal state and wind strengths is exactly as given in the book.

## 2 Exploration and learning rates

$\alpha$  is constant for the entire process and is set as 0.5.

The exploration rate  $\epsilon$  is set as 0.1 and the decay rate is set as:  $\epsilon = 0.1/(i+1)$  where  $i$  refers to the  $i$ th episode played by the agent for a particular seed value.

## 3 Plots

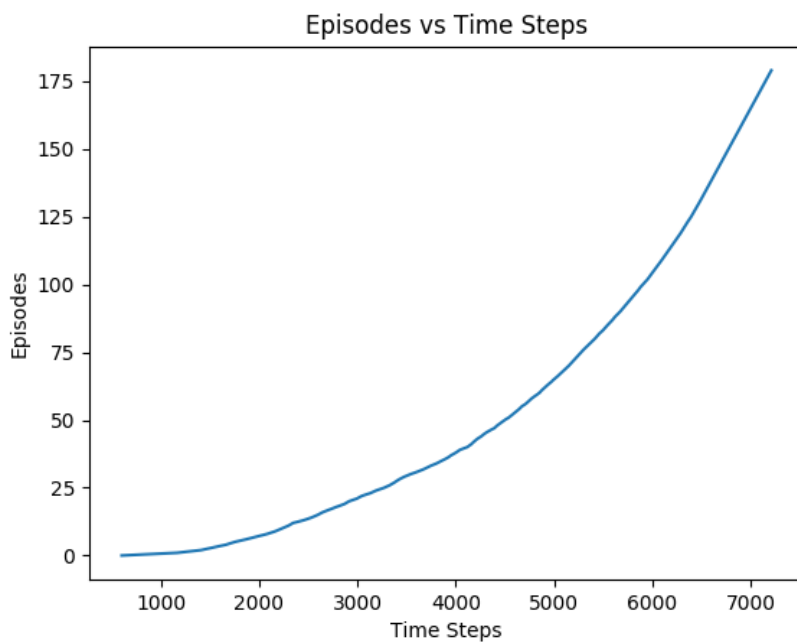


Figure 1: BaseLine Curve

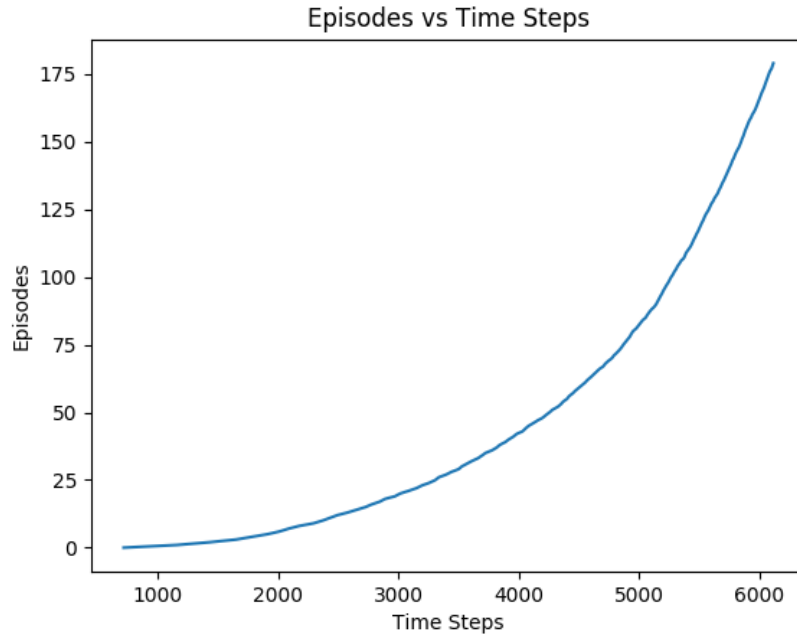


Figure 2: With King Moves

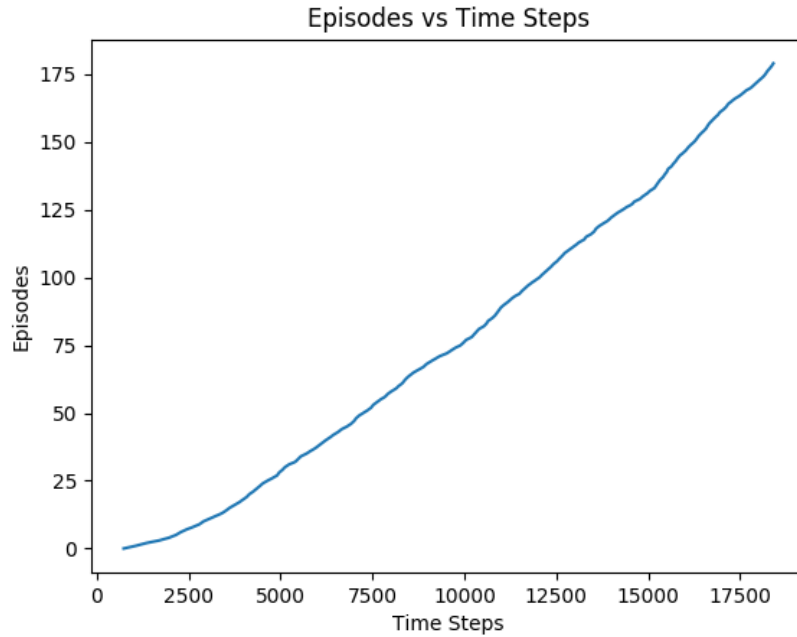


Figure 3: With Stochasticity and King Moves. Note that stochasticity has been considered even for columns where wind strength is 0, i.e., for 0 wind, the agent is equal likely to end up at the desired position (the case without stochasticity) or one cell above or one cell below the desired position.

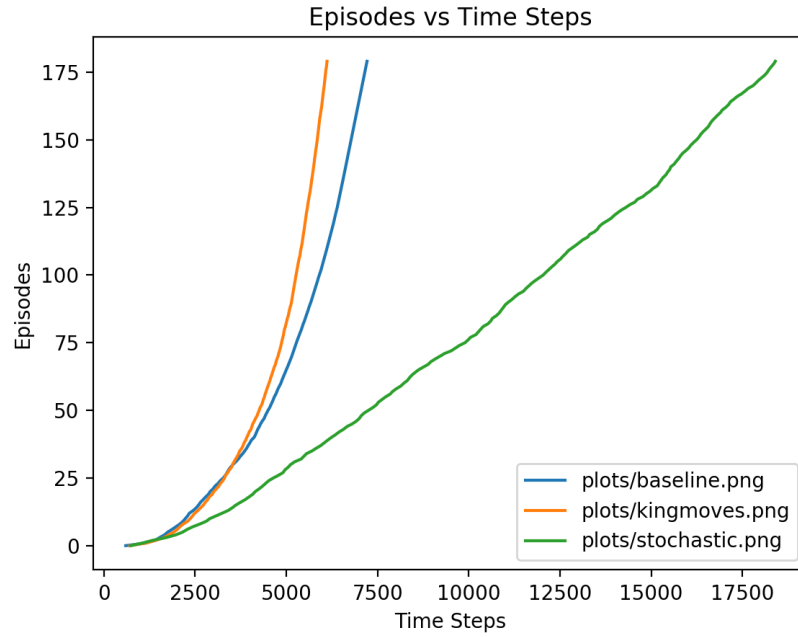


Figure 4: Comparison of all the three curves

## 4 Conclusions

1. Average Minimum steps to reach the target over 10 seed runs of 180 episodes (Average of the minimum time steps in 180 episodes for 10 seed values):
  - i Baseline: 15.0
  - ii King Moves: 7.2
  - iii Stochastic with King Moves: 7.5
2. Average Cumulative steps to reach the target over 10 seed runs of 180 episodes:
  - i Baseline: 7212.7
  - ii King Moves: 6116.4
  - iii Stochastic with King Moves: 18401.0
3. As the slope of the curve increases in figures 1,2,3 and 4, we can say that "the goal was reached more quickly over time".
4. As given in the book, the minimum number of steps to go from initial to terminal state is 15. So our baseline SARSA(0) algorithm with  $\epsilon$ -greedy (with decaying  $\epsilon$ ) algorithm works very well as it averages 15 minimum steps.
5. Adding King Moves to the environment significantly improves the performance as evidenced by the average cumulative timesteps and average minimum timestep decrease. More moves allows more freedom to the agent and the target can be reached faster.
6. As expected, on adding stochasticity makes it difficult for the agent to learn and the variance in timesteps is considerably higher (evidenced by the substantially greater cumulative time steps and almost similar minimum steps) to reach the terminal state.