

CS747 Assignment 2

Anubhav Agarwal, 17D070026

October 2020

1 Design Decisions

1.1 Value Iteration Algorithm

The iteration is carried out till the difference between successive value functions $\Sigma V(s)$ becomes less than 10^{-10} .

1.2 Howard Policy Iteration

I have used numpy to solve the set of linear equations to calculate the Bellman equations. However, the solve throws error when the matrix is singular. Which happens to be the case when running the planner in Task 2. However, Value iteration and LP work fine in Task 2. HPI also works fine in all cases for task 1.

1.3 Linear Programming

I have used the default solver in PuLP to maximize the objective function.

2 Task 2

In order to formulate the maze problem, I iterated over every row and column of the grid. each cell which was not a wall was assigned a state value.

In order to find the shortest path, I assigned the reward for every step taken as -1. The more number of steps I take, the lesser/more negative the reward becomes. When I reach the destination cell, I am providing a reward of +100 to ensure that the final destination is always considered in the solving step.(If the reward is kept same like others, the solution keeps moving back and forth between two cells in a never ending cycle).

In the decoder, since the MDP was not provided, the state action pairs were reconstructed using the grid. We use the optimal policy which was fed as an argument. Using the policy and constructed MDP, I iterate successively till I reach the end state. Every intermediate action taken is printed in terms of N/S/E/W.