Luis Berber

CSCI 166

December 11, 2020

Project

1. Domain

The domain of this project is to replicate the process that an artificial intelligence (A.I.) would take to traverse a world with trap locations, random event locations, and goal locations. There will be cells labeled “trap” that will reward an agent with a negative value if it is reached. The cells labeled “goal” will reward an agent with a positive value once it is reached. The cells labeled “randE” will not reward an agent with a value. Instead, the agent will be forced into a new cell based on a randomly generated probability and unique selection method.

1. Problem Representation

The domain of this project would consist of a Markov Decision Process (MDP) that would consists of the following components:

1. States
2. Actions
3. Rewards
4. Transition Model

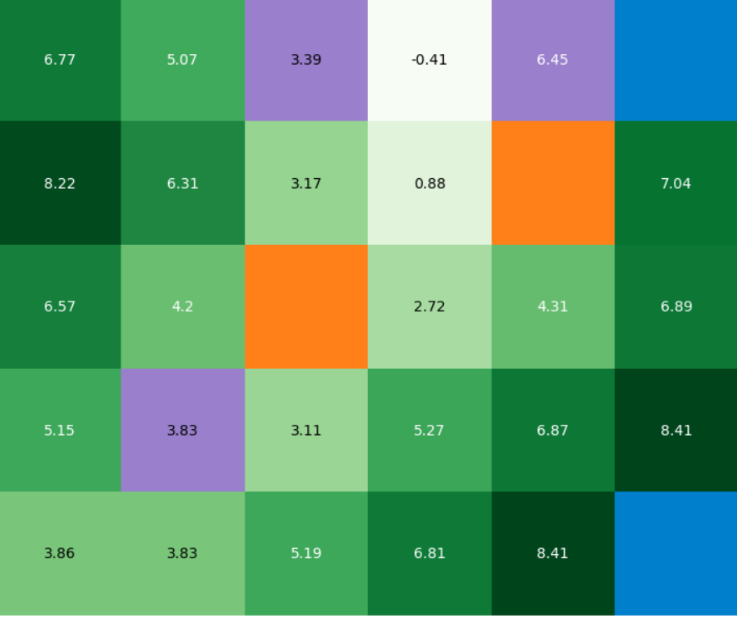
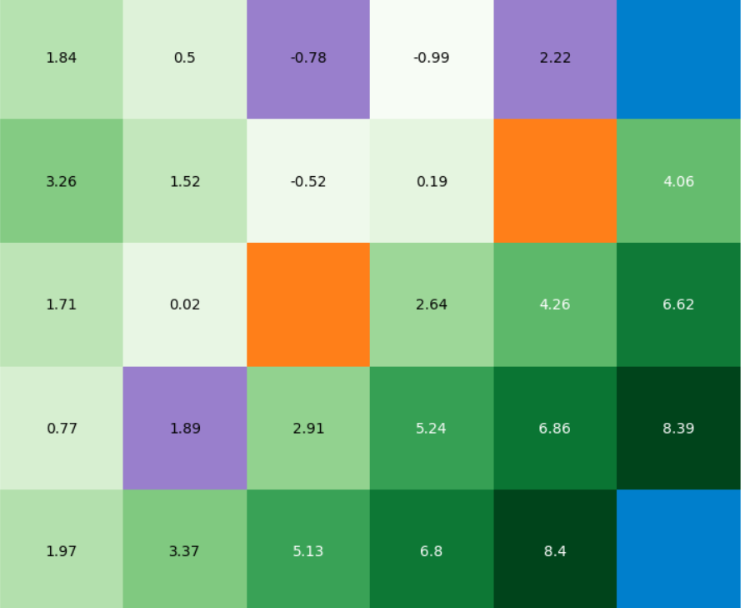
Each of these components then has its own elements. The states consist of each individual cell in the world. Those cells can then be classified further into either a “trap” state, “goal” state, “random” state and normal state. The actions consist of a set of possible moves that the agent can initiate on each cell. These moves are labeled as ‘N’, ‘S’, ‘E’ and ‘W’ indicating a move up, down, right and left, respectively. The agent has a reward element that loses value in each step and loses or gains value when it reaches a “trap” or “goal” state, respectively. The transition model determines the likelihood that the agent reaches the next state from its current state.

4. Implementation

This project was created using the Notepad ++ editor and the programming language Python 3.8.6 installed on the Windows 10 command prompt. With Python, I included the NumPy, Matplotlib and random packages. NumPy was used to handle and manipulate data in the array. Matplotlib was used for the visual representation of the world. Random was used to generate a few random values. The key algorithms that are used are value iteration and policy iteration. A link to the GitHub repository containing the project is the following: <https://github.com/angel9293/CSCI166>

5. Experimental Results

Some experiments that I made using the project involved using different sized worlds. I ran the program with a 6x5 world and a 8x7 world. The 6x5 world consists of 30 total cells of which three are “random” cells, two are “goal” cells, two are “trap” cells and 23 “normal” cells. The 8x7 world consists of 56 total cells of which three are “random” cells, three are “goal” cells, five are “trap” cells and 45 are “normal” cells. Below is a experiment using the smaller world where all the “goal” states have an equal reward value and then the values are a 1:2 proportion

(“goal” state = blue cells, “trap” state = orange cells, “random” state = purple)

As we can see, the “goal” state that is farther from the “trap” states has the highest values surrounding it while the other “goal” state has cells that lose value because they are near a “trap” state. The experiment below shows the implementation of policy iteration and shows similar results.

