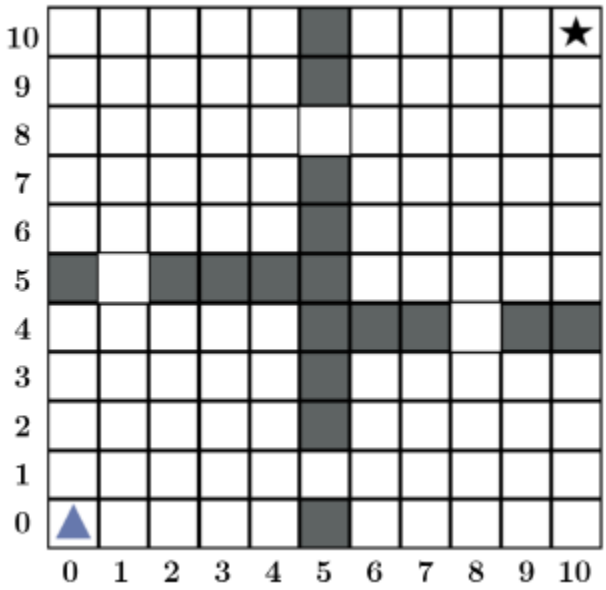
Exercise 0: An Invitation to Reinforcement Learning

Four Rooms Problem



Prof: Robert Platt

Date: September 17th, 2021

Name: Guanang Su

Part 2. Manual Policy

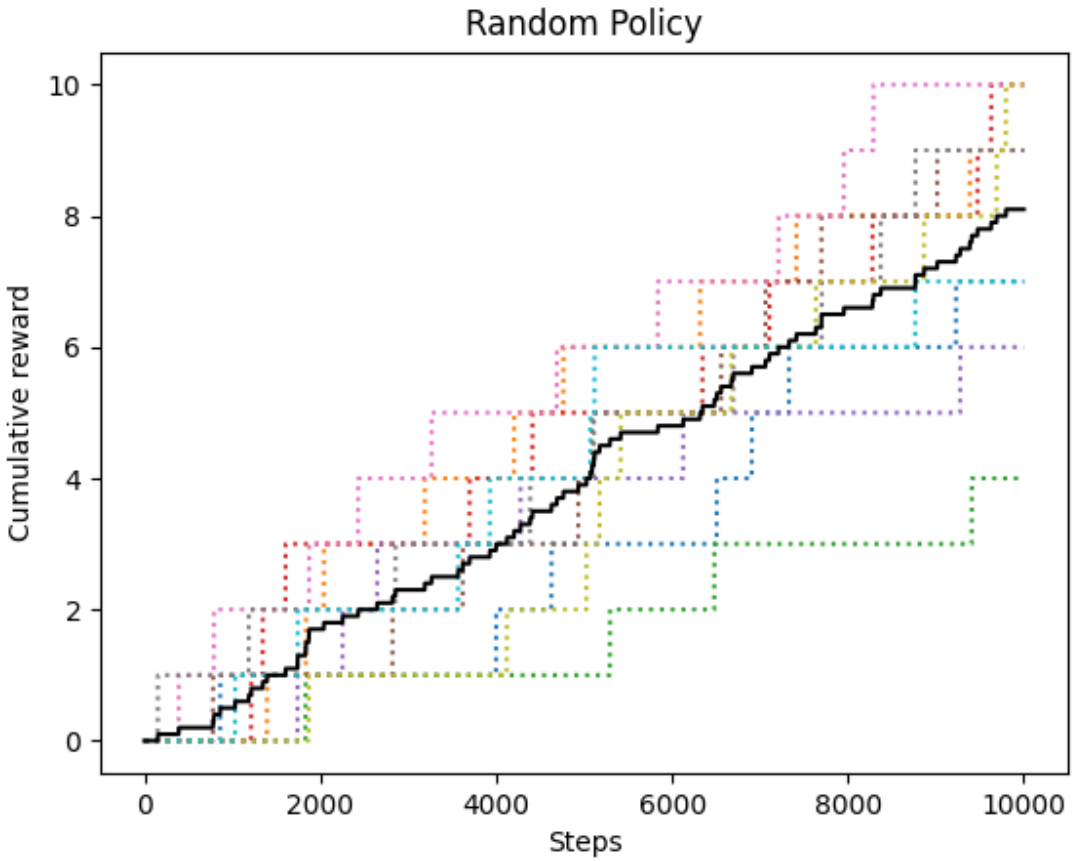
This is some procedures running in the manual policy with the user interface. The user with go from (0, 0) and try to reach the final state (10, 10). By using l(left), r(right), u(up), d(down) to control the agent to move with keyboard user input. It will show you the location after each move and the action you take after noise with a reward. After the user arrive at (10, 10), the agent will go back to the starting state and run again until you reach the maximum trails number.

```
"C:\Users\Guanang Su\AppData\Local\Programs\Python\Python38-32\python.exe" F:/CS_5180/CS5180_RL/ex0/ex0.py
This is a Four Room Problem, the initial state is (0, 0) and the final goal is (10, 10).
What policy do you want to choose?
(m: "manual", r: "random", w: "worse", b: "better", c: "comparison"): m

This is a Four Rooms Environment, you are currently at (0, 0).
What is the action you want to do?
(l: "left", d: "down", r: "right", u: "up"): u
You arrive at (0, 1) with Action.UP. The reward is 0.

This is a Four Rooms Environment, you are currently at (0, 1).
What is the action you want to do?
(l: "left", d: "down", r: "right", u: "up"): u
You arrive at (0, 2) with Action.UP. The reward is 0.

This is a Four Rooms Environment, you are currently at (0, 2).
What is the action you want to do?
(l: "left", d: "down", r: "right", u: "up"): l
You arrive at (0, 1) with Action.DOWN. The reward is 0.
```

```
This is a Four Rooms Environment, you are currently at (8, 10).
What is the action you want to do?
(l: "left", d: "down", r: "right", u: "up"): r
You arrive at (9, 10) with Action.RIGHT. The reward is 0.

This is a Four Rooms Environment, you are currently at (9, 10).
What is the action you want to do?
(l: "left", d: "down", r: "right", u: "up"): r
You did it! You arrive at (10, 10) and your reward is 1.

This is a Four Rooms Environment, you are currently at (10, 10).
What is the action you want to do?
(l: "left", d: "down", r: "right", u: "up"): r

This is a Four Rooms Environment, you are currently at (0, 0).
What is the action you want to do?
(l: "left", d: "down", r: "right", u: "up"):
```

Part 3. Random Policy

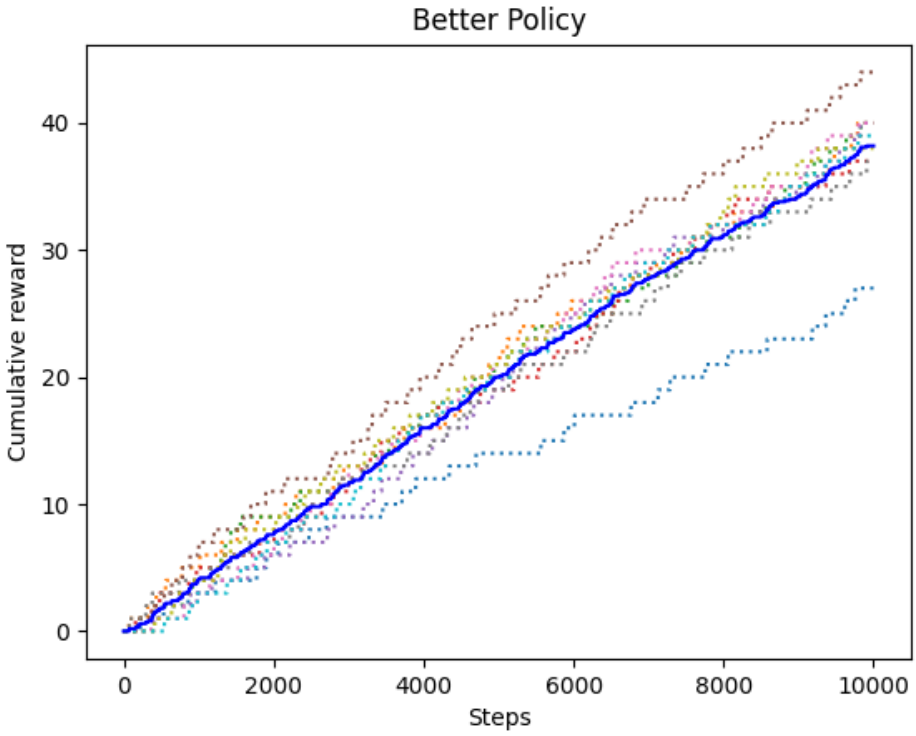
This is a figure of the random policy with running 10 trails and 10,000 steps on each trail. The trail is in dotted colored lines and perform as a black solid line with the mean of each trail on each step. The mean result of the policy achieves a cumulative reward slightly over 8, which means that on 10000 steps, it reaches the goal approximately 8 times. It takes about 125 steps to arrive the final state.

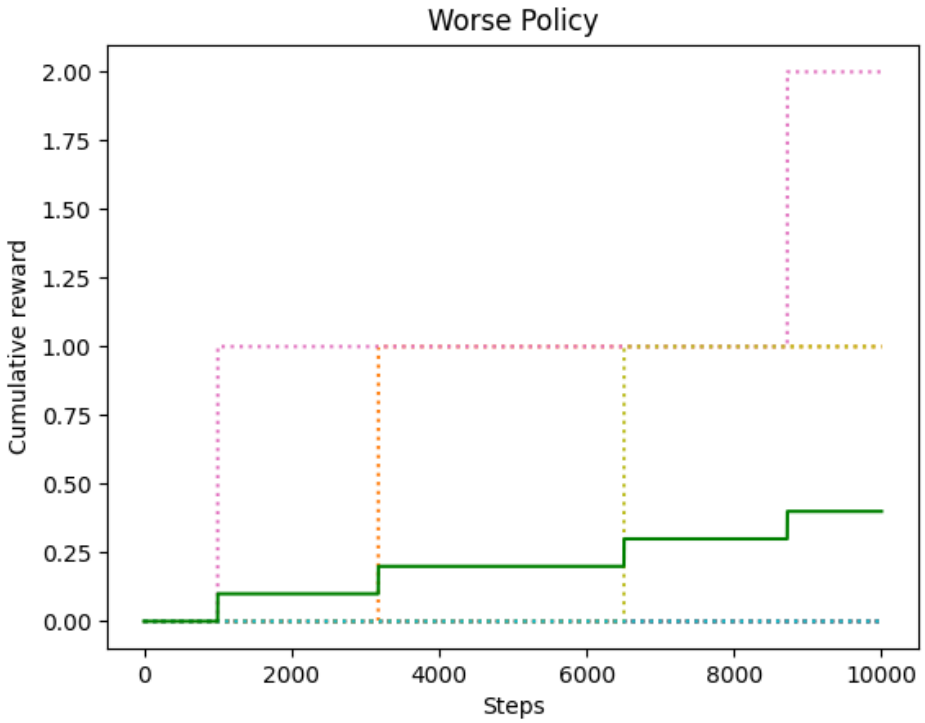


I would say that random policy is much worse than the manual policy, usually, I took about 25-30 runs to arrive at the star position, (10, 10). The reason that come up with the result in my random policy, the system picks a random action from the four direction each time from the current state, so the step is not controllable since it is random generated, and the right position is up and right on the corner. Compared to the random one, the manual policy is on the user input. User could decide their purposed action by watching the grid and wall to control the agent to move closer to the final position with a shorter path. So the manual policy would perform better than the random one.

Part 4. Two more policies and comparison between all policies

Below are two figures showing the better policy and worse policy.





For both policies, I random pick a uniform distribution of number in range of 0 to 10. And divided the probabilities of the action to go up and down. For a random one, the probability is 25% on each action for each state. To achieve a better policy with higher cumulative rewards, I increase the probability to go up and right and decrease them for the worse policy.

The detailed fraction I picked for worse is 30% left, 30% down, 20% right, 20% up.

The detailed fraction I picked for better is 17% left, 17% down, 33% right, 33% up.

By varied the probability of each action on each state, the agent is more likely to move closer or further from the initial point to the destination. The better policy applied showed a good result compared to the random one since the mean cumulative reward is about 36 compared to 8 for the random one. And mean cumulative reward for worse policy is about 0.25. In my evaluation, both policies work as expected.