CPS 570: Artificial Intelligence

Homework 5: A Markov decision process (due December 1 before class)

Please read the rules for assignments on the course web page (http://www.cs.duke.edu/courses/fall17/compsci570/). Use Piazza (preferred) or directly contact Shuzhi (shuzhiyu@cs.duke.edu), Rui-Yi (ryzhang@cs.duke.edu), or Vince (conitzer@cs.duke.edu) for any questions.

In this assignment, you will solve for the optimal policy and values of a small MDP, given by the following story. You will solve using both value iteration and policy iteration, which you are allowed to do in any way you like (by hand or computer), but you should show all your work (including your source code or whatever else you use).

A little autonomous rover on Mars depends on its solar panels for energy. Its goal is to collect as much energy as possible. It is close to a small hill; it collects the most energy when it is at the top of the hill, but it has a tendency to roll off the hill, and it takes energy to get back up the hill.

Specifically, the rover can be in one of three states: on top of the hill, rolling down the hill, or at the bottom of the hill. There are two actions that it can take in each state: drive or don't drive. Driving always costs 1 unit of energy. When it is at the top of the hill, it collects 3 units of energy; in the other two states, it collects 1 unit of energy. For example, if the rover is at the top of the hill and is driving (to stay on top of the hill), its reward is $3 - 1 = 2$.

If the rover is at the top of the hill and drives, then it is still at the top of the hill in the next period with probability .9, and rolling down in the next period with probability .1. If it is at the top of the hill and does not drive, these probabilities are .7 and .3, respectively.

If it is rolling down and drives, then with probability .3 it is at the top of the hill in the next period, with probability .6 it is still rolling down in the next period, and with probability .1 it is at the bottom in the next period. If it is rolling down and does not drive, then with probability 1 it is at the bottom in the next period.

Finally, if it is at the bottom of the hill and drives, then in the next period it is at the top of the hill with probability .6, and at the bottom with probability .4. If it does not drive, then with probability 1 it is at the bottom in the next period.

1 (25 points). Draw the MDP graphically, similarly to the machine example from class.

2 (25 points). Using a discount factor of .8, solve the MDP using value iteration (until the values have become reasonably stable). You should start with the values set to zero. You should show both the optimal policy and the optimal values.

3 (25 points). Using a discount factor of .8, solve the MDP using policy iteration (until you have complete convergence). You should start with the policy that never drives. Again, you should show both the optimal policy and the optimal values (and of course they should be the same as in **2**...).

4 (25 points). Change the MDP in three different ways: by changing the discount factor, changing the transition probabilities for a single action from a single state, and by changing a reward for a single action at a single state. Each of these changes should be performed separately starting at the original MDP, resulting in three new MDPs (which you do not have to draw), each of which is different from the original MDP in a single way. In each case, the change should be so that the optimal policy changes, and you should state what the optimal policy becomes and give a short intuitive argument for this.

Homework 5

Branch Vincent

December 1, 2017

Problem 1

The MDP is represented in Figure 1.

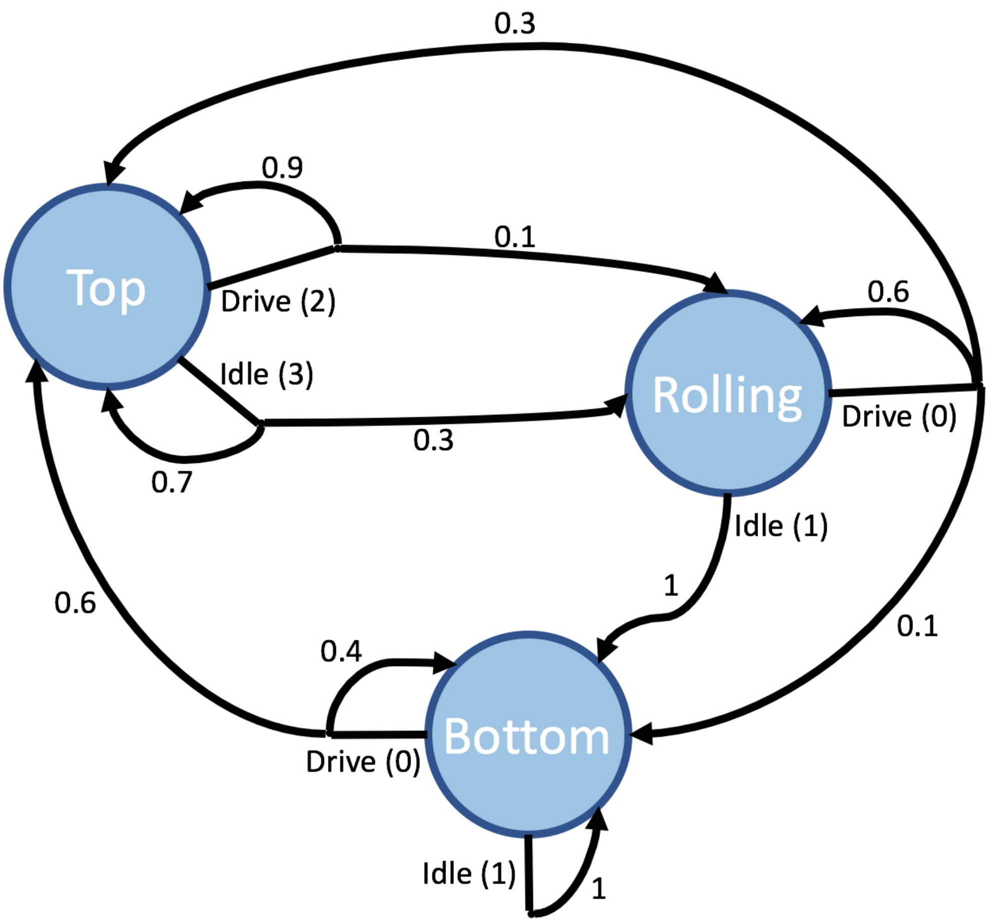

Figure 1: Markov Decision Process

Problem 2

Using value iteration and a discount factor $\delta = 0.8$, we arrive at the following optimal policy and values (given a state s).

$$\pi^*(s) = \begin{cases} \text{idle} & s = \text{top or rolling} \\ \text{drive} & s = \text{bottom} \end{cases}$$

$$v^*(s) \approx \begin{cases} 10.64 & s = \text{top} \\ 7.01 & s = \text{rolling} \\ 7.51 & s = \text{bottom} \end{cases}$$

Problem 3

Using policy iteration and $\delta = 0.8$, we arrive at the following optimal policy and values.

$$\pi^*(s) = \begin{cases} \text{idle} & s = \text{top or rolling} \\ \text{drive} & s = \text{bottom} \end{cases}$$

$$v^*(s) \approx \begin{cases} 10.64 & s = \text{top} \\ 7.01 & s = \text{rolling} \\ 7.51 & s = \text{bottom} \end{cases}$$

Problem 4

(i) Let $\delta = 0.5$. Then

$$\pi^*(s) = \begin{cases} \text{idle} & \forall\, s \end{cases}$$

This is because we now more heavily discount the future. Before at $s =$ bottom, the optimal policy was to drive in an attempt to reach the top, where we can collect a higher reward. However, since we now care less about the future, the optimal policy becomes collecting a smaller reward now.

(ii) Let the transition probability for $a =$ drive at $s =$ bottom be

$$T(\text{bottom, drive}) = \begin{cases} 0.1 & s' = \text{top} \\ 0.9 & s' = \text{rolling} \end{cases}.$$

Then

$$\pi^*(s) = \begin{cases} \text{idle} & \forall\, s \end{cases}$$

This is because, once at $s =$ bottom, we now are less likely to reach the top again. Thus, it is no longer worth trying to reach the top while collecting no reward. Instead, the optimal policy becomes collecting a smaller (but certain) reward by idling at the bottom.

(iii) Let the reward for $a =$ idle at $s =$ bottom be 3. Then

$$\pi^*(s) = \begin{cases} \text{idle} & \forall\, s \end{cases}$$

This is because the marginal benefit of driving to reach the top at $s =$ bottom is now zero, since we collect the same reward for idling at the top and at the bottom. Thus, the optimal policy becomes to always idle.

Appendix

The preceding problems were solved using the attached code.