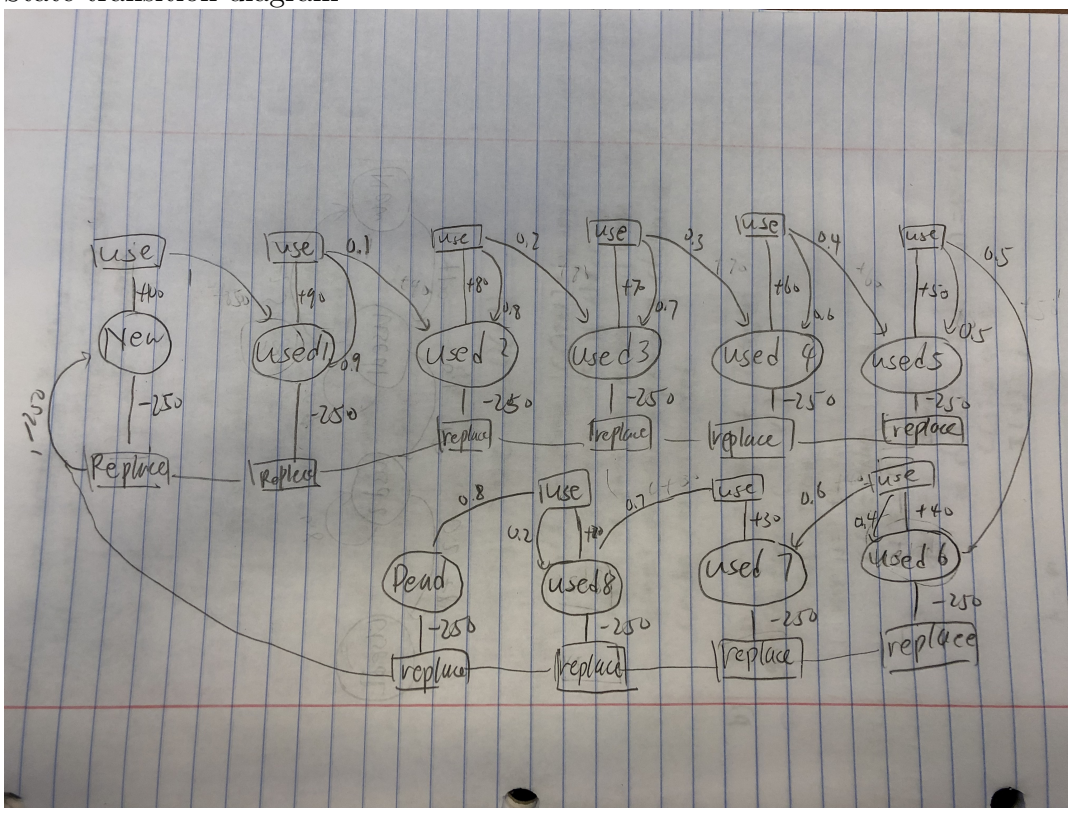
CS 520 Final: Question 2 - Markov Decision Processes

Junhan Wang

Thursday 16th May, 2019

State transition diagram



a) For each of the 10 states, what is the optimal utility (long term expected discounted value) available in that state (i.e., $U*(state)$)?

According to Bellman's Equations:

$$U^*(s) = \max_{a \in A(s)} [r_{s,a} + \beta \sum_{s'} p^a_{s,s'} U*(s')] \tag{1}$$

We can get the optimal utility equations for each state as following

1. $U^*(NEW) = \max[(100 + 0.9 * 1U^*(Used_1))]$

2. $U^*(Used_1) = \max[(90+0.9*(0.9U^*(Used_1)+0.1U^*(Used_2))), (-250+0.9*1U^*(NEW))]$

3. $U^*(Used_2) = \max[(80+0.9*(0.8U^*(Used_2)+0.2U^*(Used_3))), (-250+0.9*1U^*(NEW))]$

4. $U^*(Used_3) = \max[(70+0.9*(0.7U^*(Used_3)+0.3U^*(Used_4))), (-250+0.9*1U^*(NEW))]$

5. $U^*(Used_4) = \max[(60+0.9*(0.6U^*(Used_4)+0.4U^*(Used_5))), (-250+0.9*1U^*(NEW))]$

6. $U^*(Used_5) = \max[(50+0.9*(0.5U^*(Used_5)+0.5U^*(Used_6))), (-250+0.9*1U^*(NEW))]$

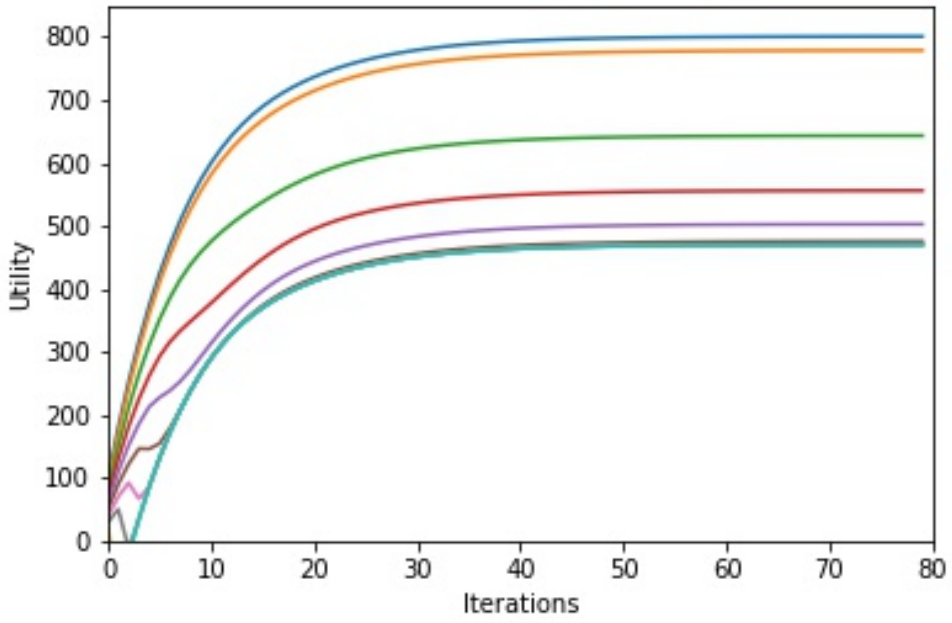
7. $U^*(Used_6) = \max[(40+0.9*(0.4U^*(Used_6)+0.6U^*(Used_7))), (-250+0.9*1U^*(NEW))]$

8. $U^*(Used_7) = \max[(30+0.9*(0.3U^*(Used_7)+0.7U^*(Used_8))), (-250+0.9*1U^*(NEW))]$

9. $U^*(Used_8) = \max[(20+0.9*(0.2U^*(Used_8)+0.8U^*(Dead))), (-250+0.9*1U^*(NEW))]$

10. $U^*(Dead) = -250 + 0.9 * 1U^*(NEW))$

Suppose the optimal utilities of each state are 0 at the beginning, we get the graph of utilities of each states over 80 iterations.



We can see that after 5 iterations, $U*(Used6)$, $U*(Used7)$, $U*(Used8)$, $U*(dead)$ begin to get same value. That means after 5 iteration the robot prefer to replace machine at used6 state, since that will get higher utility. After 50 iterations the utility of each state are converged and will not have big change any more. I would say that should be the optimal utilities of each state. Here are optimal utilities of each state after 80 iterations

```
u_new:  800.4317215421976
u_used1:  778.2690026514388
u_used2:  643.1271786944175
u_used3:  556.0305273768115
u_used4:  502.74431291683374
u_used5:  475.7553132243413
u_used6:  470.38854938797783
u_used7:  470.38854938797783
u_used8:  470.38854938797783
u_dead:  470.38854938797783
```

b) What is the optimal policy that gives you this optimal utility - i.e., in each state, what is the best action to take in that state? According to optimal equation:

$$\pi^*(s) = argmax_{a \in A(s)}[r_{s,a} + \beta \sum_{s'} p^a_{s,s'} U * (s')] \tag{2}$$

We can get the optimal policy equations for each state as following:

1. $\pi^*(NEW) = argmax[(100 + 0.9 * 1U^*(Used_1))]$
2. $\pi^*(Used_1) = argmax[(90 + 0.9 * (0.9U^*(Used_1) + 0.1U^*(Used_2))), (-250 + 0.9 * 1U^*(NEW))]$
3. $\pi^*(Used_2) = argmax[(80 + 0.9 * (0.8U^*(Used_2) + 0.2U^*(Used_3))), (-250 + 0.9 * 1U^*(NEW))]$
4. $\pi^*(Used_3) = argmax[(80 + 0.9 * (0.7U^*(Used_3) + 0.3U^*(Used_4))), (-250 + 0.9 * 1U^*(NEW))]$
5. $\pi^*(Used_4) = argmax[(80 + 0.9 * (0.6U^*(Used_4) + 0.4U^*(Used_5))), (-250 + 0.9 * 1U^*(NEW))]$
6. $\pi^*(Used_5) = argmax[(80 + 0.9 * (0.5U^*(Used_5) + 0.5U^*(Used_6))), (-250 + 0.9 * 1U^*(NEW))]$
7. $\pi^*(Used_6) = argmax[(80 + 0.9 * (0.4U^*(Used_6) + 0.6U^*(Used_7))), (-250 + 0.9 * 1U^*(NEW))]$
8. $\pi^*(Used_7) = argmax[(80 + 0.9 * (0.3U^*(Used_7) + 0.7U^*(Used_8))), (-250 + 0.9 * 1U^*(NEW))]$
9. $\pi^*(Used_8) = argmax[(80 + 0.9 * (0.2U^*(Used_8) + 0.8U^*(Dead))), (-250 + 0.9 * 1U^*(NEW))]$
10. $\pi^*(Dead) = argmax[-250 + 0.9 * 1U^*(NEW)]$

The optimal policy for each state by above optimal utilities are

```
pi_new:  use
pi_used1:  use
pi_used2:  use
pi_used3:  use
pi_used4:  use
pi_used5:  use
pi_used6:  repalce
pi_used7:  repalce
pi_used8:  repalce
pi_dead:  replace
```

Since there is only one action can be taken is dead state, which is 'replace', the optimal policy of dead state have to be 'replace'.

c) Instead of buying a new machine, a MachineSellingBot offers you the following option: you could buy a used machine, which had an equal chance of being in Used1 and Used2. Intuitively:

– If the MachineSellingBot were offering you this option for free, you would never buy a new machine.
– If the MachineSellingBot were offering you this option at a cost of 250, you would never take this option over buying a new machine.

What is the highest price for which this used machine option would be the rational choice? i.e., what price should MachineSellingBot be selling this option at?

Since buying a used or new machine always bring the cast, the optimal policies of every states until dead state are 'use'. The optimal policy is to replace the machine at Used6 state. Then, we only need to consider buying a used machine at Used6 state. We can just add an action of buying a used machine at dead state. Since the value of action 'use' is lower than 'repalce', we do not need to consider 'use'. The highest price of the used machine should make the optimal value of buying a used machine equal to the optimal value of buying a new machine. Then, we get a new optimal policy equation of dead state.
$\pi^*(Used6) = argmax[(80{+}0.9{*}(0.4U^*(Used_6){+}0.6U^*(Used_7))), (-250{+}0.9{*}1U^*(NEW)), -price{+}0.9 * (0.5 * U^*(Used1) + 0.5 * U^*(Used2))]$

Makes value of buying a new machine equals to the value of buying a used machine
$-250 + 0.9 * 1U^*(NEW) = -price + 0.9 * (0.5 * U^*(Used1) + 0.5 * U^*(Used2))$
By putting the optimal utility of state used1 ad used2 that we just computed into the equation. We can get the highest rational price of a used machine is: $price = 169.24$

d) For different values of β (such that 0 ¡ β ¡ 1), the utility or value of being in certain states will change. However, the optimal policy may not. Compare the optimal policy for $\beta = 0.1, 0.3, 0.5, 0.7, 0.9, 0.99$, etc. Is there a policy that is optimal for all sufficiently large β? Does this policy make sense? Explain.

For $\beta = 0.1, 0.3, 0.5, 0.7$
$\pi^*(New) = use$
$\pi^*(Used1) = use$
$\pi^*(Used2) = use$
$\pi^*(Used3) = use$
$\pi^*(Used4) = use$
$\pi^*(Used5) = use$
$\pi^*(Used6) = use$
$\pi^*(Used7) = use$
$\pi^*(Used8) = replace$
$\pi^*(dead) = replace$

For $\beta = 0.9$
$\pi^*(New) = use$
$\pi^*(Used1) = use$
$\pi^*(Used2) = use$
$\pi^*(Used3) = use$
$\pi^*(Used4) = use$
$\pi^*(Used5) = use$
$\pi^*(Used6) = replace$
$\pi^*(Used7) = replace$
$\pi^*(Used8) = replace$
$\pi^*(dead) = replace$

For $\beta = 0.99$
$\pi^*(New) = use$
$\pi^*(Used1) = use$
$\pi^*(Used2) = use$
$\pi^*(Used3) = use$
$\pi^*(Used4) = replace$
$\pi^*(Used5) = replace$
$\pi^*(Used6) = replace$
$\pi^*(Used7) = replace$
$\pi^*(Used8) = replace$
$\pi^*(dead) = replace$

When β? equals 0.9999 , the optimal policy is to replace the machine at state used 3. So when can know that replacing machine at state used 3 is always optimal for all all sufficiently large β, since it can the optimal utility of 'replace' can not higher than the optimal utility of 'use' at state used1, used2.