# CS360 – Homework #13

Markov Decision Processes

1) Invent a simple Markov decision process (MDP) with the following properties:
a) it has a goal state, b) its immediate action costs are all positive, c) all of its
actions can result with some probability in the start state, and d) the optimal
policy without discounting differs from the optimal policy with discounting and
a discount factor of 0.9. Prove d) using value iteration.
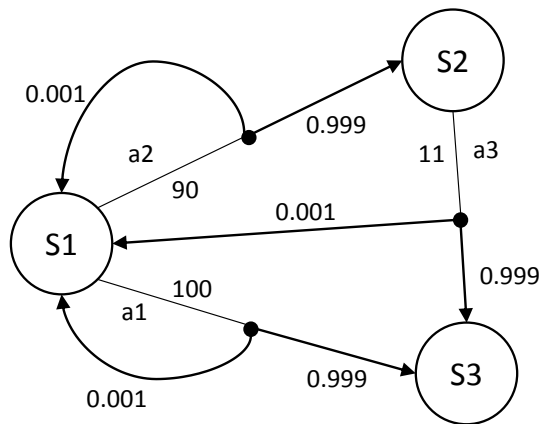
With no discount factor, action a1 is preferred over action a2 in state s1:

	i	0	1	2	3	4	...
State1 a1	0	100	**100.09**	**100.10009**	**100.1001001**	
a2	0	**90**	101.079	101.179	101.18909	
State2 a3	0	11	11.09	11.10009	11.10010009	
State3		0	0	0	0	0	

With discount factor = 0.9, action a2 is preferred over action a1 in state s1:

	i	0	1	2	3	4	...
State1 a1	0	100	100.081	100.089974	100.0900476	
a2	0	**90**	**99.9711**	**100.0529011**	**100.0610432**	
State2 a3	0	11	11.081	11.08997399	11.09004761	
State3		0	0	0	0	0	

2) Consider the following problem (with thanks to V. Conitzer): Consider a rover
that operates on a slope and uses solar panels to recharge. It can be in one of
three states: high, medium and low on the slope. If it spins its wheels, it climbs
the slope in each time step (from low to medium or from medium to high) or
stays high. If it does not spin its wheels, it slides down the slope in each time
step (from high to medium or from medium to low) or stays low. Spinning its

wheels uses one unit of energy per time step. Being high or medium on the slope gains three units of energy per time step via the solar panels, while being low on the slope does not gain any energy per time step. The robot wants to gain as much energy as possible.

a) Draw the MDP graphically. b) Solve the MDP using value iteration with a discount factor of 0.8. c) Describe the optimal policy.
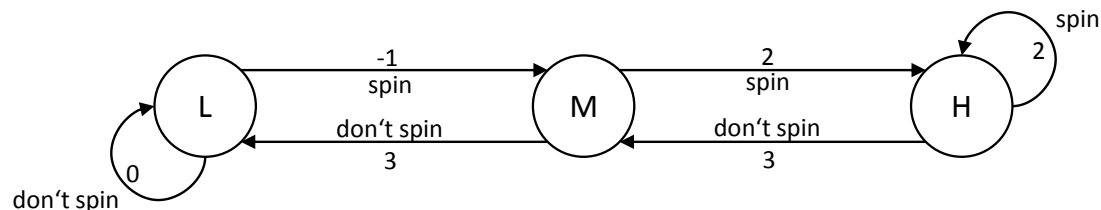
where L = low, M = medium and H = high.

Starting with 0 as initial values, value iteration calculates the following:

	L		M		H	
ITR	spin	don't	spin	don't	spin	don't
1	−1.00	0.00*	2.00	3.00*	2.00	3.00*
2	1.40*	0.00	4.40*	3.00	4.40	5.40*
3	2.52*	1.12	6.32*	4.12	6.32	6.52*
4	4.06*	2.02	7.22*	5.02	7.22	8.06*
5	4.77*	3.24	8.44*	6.24	8.44	8.77*
6	5.76*	3.82	9.02*	6.82	9.02	9.76*
7	6.21*	4.60	9.80*	7.60	9.80	10.21*
8	6.84*	4.97	10.17*	7.97	10.17	10.84*
9	7.14*	5.47	10.67*	8.47	10.67	11.14*
10	7.54*	5.71	10.91*	8.71	10.91	11.54*
...						
20	8.64*	6.88	12.08*	9.88	12.08	12.64*
...						
28	8.76*	7.00	12.20*	10.00	12.20	12.76*
29	8.76*	7.00	12.20*	10.00	12.20	12.76*

At each iteration, the value of a state is the value of the maximizing action in that state (since we are trying to maximize energy) and is marked with an asterisk. For instance, in iteration 4, the value of L, $v_4(L)$, is computed as follows:

$v_4(L, spin) = 0.8 \times v_3(M) - 1 = 0.8 \times 6.32 - 1 \sim 4.06$

$v_4(L, don't) = 0.8 \times v_3(L) + 0 = 0.8 \times 2.52 \sim 2.02$

$v_4(L) = max(v_4(L, spin), v_4(L, don't)) = 4.06$

The optimal policy is to spin when the rover is low or medium on the slope and not to spin when it is high on the slope.

Now answer the three questions above for the following variant of the robot problem: If it spins its wheels, it climbs the slope in each time step (from low to medium or from medium to high) or stays high, all with probability 0.3. It stays where it is with probability 0.7. If it does not spin its wheels, it slides down the slope to low with probability 0.4 and stays where it is with probability 0.6. Everything else remains unchanged from the previous problem.
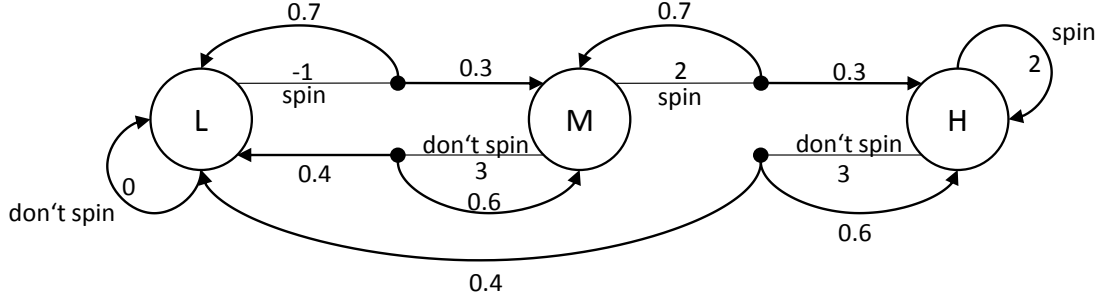
Starting with 0 as initial values, value iteration calculates the following:

	L		M		H	
ITR	spin	don't	spin	don't	spin	don't
1	-1.00	0.00*	2.00	3.00*	2.00	3.00*
2	-0.28	0.00*	4.40	4.44*	4.40	4.44*
3	0.07*	0.00	5.55*	5.13	5.55*	5.13
4	0.37*	0.05	6.44*	5.69	6.44*	5.69
5	0.75*	0.30	7.15*	6.21	7.15*	6.21
6	1.14*	0.60	7.72*	6.67	7.72*	6.67
7	1.49*	0.91	8.18*	7.07	8.18*	7.07
8	1.80*	1.19	8.54*	7.40	8.54*	7.40
9	2.06*	1.44	8.83*	7.68	8.83*	7.68
10	2.27*	1.65	9.07*	7.90	9.07*	7.90
...						
20	3.08*	2.45	9.90*	8.72	9.90*	8.72
...						
30	3.17*	2.53	9.99*	8.81	9.99*	8.81
31	3.17*	2.54	9.99*	8.81	9.99*	8.81
32	3.17*	2.54	9.99*	8.81	9.99*	8.81

In this variant, in iteration 4, the value of L, $v_4(L)$, is computed as follows:

$v_4(L, spin) = 0.8 \times (0.3 \times v_3(M) + 0.7 \times v_3(L)) - 1$

$= 0.8 \times (0.3 \times 5.55 + 0.7 \times 0.07) - 1 \sim 0.37$

$v_4(L, don't) = 0.8 \times v_3(L) + 0 = 0.8 \times 0.07 = 0.056$

$v_4(L) = max(v_4(L, spin), v_4(L, don't)) = 0.37$

The optimal policy is to spin, wherever the rover is on the slope.

3) You won the lottery and they will pay you one million dollars each year for 20 years (starting this year). If the interest rate is 5 percent, how much money do you need to get right away to be indifferent between this amount of money and the annuity?

A million dollars we get right away is worth a million dollars to us now. A million dollars we get in a year from now is worth $\gamma = 1/(1 + 0.05)$ million dollars to us now because, with interest, it would be $(1/1.05) \times 1.05 = 1$ million dollars in a year. Similarly, a million dollars we get in 19 years from now (in the beginning of the 20th year) is worth only $(1/1.05)^{19} \sim 0.4$ million dollars to us now. Therefore, getting paid a million dollars each year for 20 years is worth $1 + \gamma + \gamma^2 + \cdots + \gamma^{19} = (1 - \gamma^{20})/(1 - \gamma) \sim 0.623/0.0476 \sim 13.08$ million dollars to us now.

4) Assume that you are trying to pick up a block from the table. You drop it accidentally with probability 0.7 while trying to pick it up. If this happens, you try again to pick it up. How many attempts does it take on average before you pick up the block successfully?

We can model this problem as an MDP as follows:
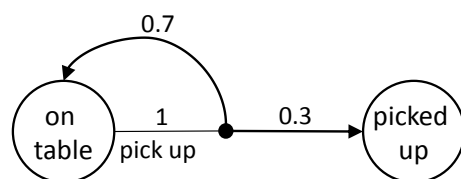
Now, we can calculate the expected cost with $X = 1 + 0.7X + 0.3Y$ and $Y = 0$ (where $X =$ on table and $Y =$ picked up), which gives us $X = 1/0.3 \sim 3.33$.

The Markov assumption is the assumption that, whenever one executes an action in a state, the probability distribution over its outcomes is the same, no matter how one got into the state. Is this assumption realistic here?

No. If the pickup operation does not work the first time (e.g. because the block is too slippery), it will likely not work the next time either.

5) Assume that you use undiscounted value iteration (that is, value iteration with a discount factor of 1) for a Markov decision process with goal states, where the action costs are greater than or equal to zero. Give a simple example that shows that the values that value iteration converges to can depend on the initial values of the states, in other words, the values that value iteration converges to are not necessarily equal to the expected goal distances of the states.
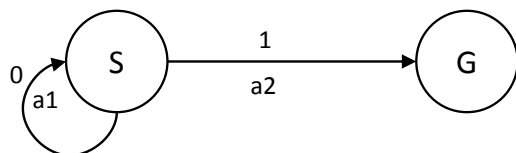
Consider the initial values $v^0(S) = 0$ and $v^0(G) = 0$. Value iteration determines the values after convergence to be $v^*(S) = 0$ and $v^*(G) = 0$, yet the (ex-

pected) goal distance of S is 1, not 0. Now consider the initial values $v^0(S) = 2$ and $v^0(G) = 0$. Value iteration determines the values after convergence to be $v^*(S) = 1$ and $v^*(G) = 0$.