Multi-Agent Systems

Homework Assignment 3

Martynas Vaznonis (2701013)
m.vaznonis@student.vu.nl

1 Bellman equations

The Bellman equations for v_π and q_π are $v_\pi(s) = \sum_a \pi(a|s) \sum_{s'} p(s'|s,a)[r(s,a,s') + \gamma v_\pi(s')]$ and $q_\pi(s,a) = \sum_{s'} p(s'|s,a)[r(s,a,s') + \gamma \sum_{a'} \pi(a'|s') q_\pi(s',a')]$ respectively.

1. If the policy is deterministic, we can express selected action as $a_s = \pi(s)$. Thus, the deterministic Bellman equations are $v_\pi(s) = \sum_{s'} p(s'|s,a_s)[r(s,a_s,s') + \gamma v_\pi(s')]$ and $q_\pi(s,a) = \sum_{s'} p(s'|s,a)[r(s,a,s') + \gamma q_\pi(s',a_{s'})]$.

2. For simplicity I will express the new state as a function of the old state and action $s' = s'(s,a) = s'(s,\pi(s))$. Then, the Bellman equations are $v_\pi(s) = r(s,a,s') + \gamma v_\pi(s')$ and $q_\pi(s,a) = r(s,a,s') + \gamma q_\pi(s',a_{s'})$.

2 MDP 1

1. Without any discounting, the value functions will always be 10, $\forall s : v_\pi(s) = 10$ and $\forall s,a : q_\pi(s,a) = 10$.

2. All policies would be equal and optimal with optimal value functions $\forall s : v^*(s) = 10$ and $\forall s,a : q^*(s,a) = 10$.

3. In such a case, every step incurs a penalty. So the optimal policy would make a beeline for the terminal state. Because there are an even number of non-terminal states, the agent is always at most $\frac{n-1}{2}$ steps away from the terminal state and there is no node v where the agent is equidistant from the terminal node. This means that the optimal policy is unique. The optimal state value function is $v^*(s) = 11 - d(s)$, where $d(s)$ is number of steps from the terminal state (1 at minimum). The optimal state-action value function is best expressed in terms of the optimal state value function $q^*(s,a) = \begin{cases} v^*(s') - 1 & \text{if } s,a \to s' \wedge s' \text{ is not terminal} \\ 10 & \text{if } s,a \to s' \wedge s' \text{ is terminal} \end{cases}$

4. The reasoning is similar as in (3). Thus, there is one unique best policy which takes the quickest path to the terminal state. Optimal state value function is $v^*(s) = 10\gamma^{d(s)-1}$. The optimal state-action value function is best expressed in terms of the optimal state value function $q^*(s,a) = \begin{cases} v^*(s') & \text{if } s,a \to s' \wedge s' \text{ is not terminal} \\ 10 & \text{if } s,a \to s' \wedge s' \text{ is terminal} \end{cases}$

5. The optimal policy would no longer be unique as at the most distant node (node $\frac{n+1}{2}$) from the terminal node the choice of action would be arbitrary. That is all policies $\pi^*(a|\frac{n+1}{2}) = p$ would be optimal regardless of the value of p, if $0 \leq p \leq 1$. The optimal value functions are the same as in (3).

3 MDP 2

1. Value evaluation can be used to find the answer and evaluate the policy π. First, all values are initialized as $\forall s : v(s) = 0$. Then, all nodes are swept over and updated such that $v(s) = \sum_a \pi(a|s)[r(s,a,s') + v(s')] = \frac{1}{2}\sum_a [r(s,a,s') + v(s')]$. Then, for example, $v(1) = \frac{20+0}{2} = 10$, $v(2) = \frac{10+0}{2} = 5$, and so on. Then this process is repeated until convergence. The final values can be seen below.

$$v(1) = 15 \qquad v(2) = 10 \qquad v(3) = 5 \qquad v(4) = 5 \qquad v(5) = 10 \qquad v(6) = 15$$

2. Because there is no discounting nor transition penalty, an optimal policy would select arbitrary actions with at least some probability of the agent transitioning into the terminal state A and with a 0 probability of transitioning into state B. The value function then is $\forall s : v^*(s) = 20$.

3. In this case the agent is incentivized to reach state A as quickly as possible so the optimal policy is unique. Then the value of state is higher the closer to state A that it is, with the values written below.

$$v(1) = 20 \qquad v(2) = 19 \qquad v(3) = 18 \qquad v(4) = 18 \qquad v(5) = 19 \qquad v(6) = 20$$

4. The reasoning here is similar as in (3), but the states close to terminal state B (states 3 and 4) are indifferent in which action to take. So there are infinitely many optimal policies with arbitrary probability p when choosing between the two available actions. The state value function values can be seen below.

$$v(1) = 20 \qquad v(2) = 10 \qquad v(3) = 0 \qquad v(4) = 0 \qquad v(5) = 10 \qquad v(6) = 20$$

4 MDP 3

Questions:

- To find the optimal policy, the optimal value function can be first computed with value iteration, and the optimal policy derived greedily. The optimal value function is $v^*(s_1) \approx 52.63$ and $v^*(s_2) \approx 47.37$ if a_2 is selected from s_1 and a_2 is selected from s_2. Thus, that is the optimal policy as given by value iteration.

- As seen in the previous subquestion, the optimal state value for s_2 is $v^*(s_2) \approx 47.37$. It can be derived iteratively or analytically in this case. The latter is $v^*(s_2) = \gamma v^*(s_1) = \gamma[10+\gamma v^*(s_2)] \Rightarrow v^*(s_1) = \frac{10\gamma}{1-\gamma^2} \approx 47.37$.

- If the γ is set to a low value, it means we care very little about rewards in future states and become more greedy. For example, if the discount factor is very small $\gamma = 10^{-3}$, then in state s_2

it is no longer optimal to take action a_2. The future reward of 10 is so down-weighed that the immediate reward of 1 is preferred, changing the optimal policy.