QUIZ 4 - MDPs and the Bellman Equation

Hoang-Dung Bui,
George Mason University
Fairfax, USA
hbui20@gmu.edu

I. Relationships between MDP quantities

Question 1: Define the optimal value function $V^*(s)$ in terms of the optimal state-action value function $Q^*(s,a)$.
Answer: The optimal value function $V^*(s)$:

$$v^*(s) = max_a \mathbf{E}[R_{t+1} + \gamma v^*(S_{t+1})|S_t = s, A_t = a] = max_a \sum_{s',r} p(s', r|s, a)[r + \gamma v^*(s')]$$

$V^*(s)$ returns the maximum expected value in the possible action set of A at state s, so we can determine the relation between optimal value function and optimal action value function as following:

$$v^*(s) = max_{a \in A} \quad q^*(s,a)$$

Question 2: Imagine you have access only to the optimal state-action value function $Q^*(s,a)$. Write the optimal policy $\pi^*(s)$ in terms of Q^*.
Answer:

$$\pi^*(s) = argmax_{a \in A} \quad q^*(s,a)$$

Question 3: Now we'll go the other way: imagine you have V^*, but you want Q^*. Using the MDP model (including the reward and the state transition model), write Q^* in terms of V^*.
Answer:

$$\begin{aligned} q^*(s,a) &= max_\pi q_\pi(s,a) \\ &= \mathbf{E}[R_{t+1} + \gamma max_{a'} q^*(S_{t+1}, a')|S_t = s, A_t = a] \\ &= \sum_{s',r} p(s', r|s, a)[r + \gamma max_{a'} q^*(s', a')] \\ &= \sum_{s',r} p(s', r|s, a)r + \gamma \sum_{s',r} p(s', r|s, a) max_{a'} q^*(s', a') \\ &= R_s^a + \gamma \sum_{s' \in S} P(s'|s, a)v^*(s') \end{aligned}$$

Question 4: Given the optimal value function $V^*(s)$, write a definition of $\pi^*(s)$ (without using Q^*). You may use your answer to part 3 in your response
Answer:

$$\begin{aligned} \pi^*(s) &= argmax_{a \in A} \quad q^*(s,a) \\ &= argmax_{a \in A} \quad R_s^a + \gamma \sum_{s' \in S} P(s'|s, a)v^*(s') \end{aligned}$$

II. Describing an MDP for Pac-Man

In this task, we will convert the Pac-Man video game into an MDP.

1) Describe the state space of Pac-Man: The board, the pellets' locations, the pellets' states (exit or eaten), the ghosts' location, the ghosts' states (normal or blue or die), count down's counter (for the ghost in blue state - value could be negative, zero, and positive), number of life left, the dots' location, the dots' states (eaten or not), the Pac-Man's location, the Pac-Man's points, the Pac-Man's state (die or alive), number of dots were eaten, number of fruits.
2) There are five actions for the Pac-Man: move left, move right, move down, move up, and eat.
3) Describe the transition model: All the items and the behaviors in the game are deterministic except the movement of the ghosts (randomness), so the transition model are randomness.
   a) As Pac-Man makes one move, it will change its locations, so we have a new state. If the count down's counter is negative, do nothing more. If it is positive, reduce the counter by 1. If it is zero, change the stage of the ghosts into normal, and assigned the counter = -1.
   b) At the new position, if one of the ghost is there and its state is normal, Pac-Man will die. It changes the state of Pac-Man and reduce the number of life by 1. If number of life is smaller than zero, the game's over. If no, the game will be reset, and Pac-Man will be initialized at the predefined location. If the ghosts' state is blue, Pac-Man can eat it and increase its point. The ghost state changes to die, and regenerated a new one.
   c) If there is dot at the new position, Pac-Man eat it, and increase its point by 1. The dot's state at that position will be changed to eaten. If the point is larger than or equal to a threshold, the new point = the point - threshold and increase 1 fruit. If the count down's counter is negative, do nothing more. If it is positive, reduce the counter by 1. If it is zero, change the stage of the ghosts into normal, and assigned the counter = -1.
   d) If there is a pallet at the position, Pac-Man will eat it and increase some bonus for its point. If the point is larger than or equal to a threshold, the new

point = the point - threshold and increase 1 more fruit. Then it changes the ghosts' states into *blue*, and set the count down's clock = limit time.

e) If all the dots' states are eaten, the game is over and Pac-Man win.

4) Define the reward function: The reward functions inputs are: pac-man's location, and are there dot, pellet, blue ghost, and normal's ghost at the location.
   a) If dot is available, the reward is 1.
   b) If a pellet is available, the reward is 0.
   c) If a blue ghost is available, the reward is 10.
   d) If a normal ghost is available, the reward is -20.