Reinforcement Learning

Student name and id: Nicklas Hansen (s153077)

Chapter: 3

Exercise 3.2

A MDP assumes that its states have the *Markov property*, i.e. information about all aspects of the past agent-environment interaction that makes a difference for the future. This may not always be satisfied in real-life examples, as the environment can be very complex to model and may contain a high degree of uncertainty and unknowns (even in the past).
Another significant limitation of a MDP is that the agent only can be rewarded with a scalar, single-value reward. As such, if we were to model a problem with two reward signals, the agent would likely fail to optimize very well.
Additionally, a MDP requires that all states and all actions in a given state can be formulated precisely and uniquely. So in the case where the number of states or actions are either unknown or infinite, the MDP framework cannot be applied properly.

Exercise 3.3

The line between agent and environment should be drawn such that the specific goal can be clearly defined in terms of state-action sequences. In the case of autonomous driving, actions would be to control the steering wheel, accelerator etc. of the car if the goal is to just drive.
If the goal is to drive to a specific location, then control of the individual car components become a sub-goal and the agent-environment separation should thus happen at a higher level. In some cases, one can imagine that the system may actually be split into two separate agents working together by acting at different levels of abstraction.

Exercise 3.4

The new table is given as follows:

s	a	s'	r	$p(s', r\|s, a)$
high	search	high	r_{search}	α
high	search	low	r_{search}	$1 - \alpha$
low	search	high	r_{search}	$1 - \beta$
low	search	low	-3	β
high	wait	high	r_{wait}	1
low	wait	low	r_{wait}	1
low	recharge	high	0	1

Exercise 3.6

In both scenarios, the return could be considered a measure of the likelihood (*not* probability) that the pole will fall, e.g. -0.8 estimates that the pole is falling very soon, whereas -0.08 estimates that the pole is unlikely to fall in coming time steps.
It is assumed that $\gamma < 1$. In the episodic case, the discounting is dependent on the finite number of time steps T, whereas the discounted, continuing formulation of the task is a sum of infinite terms.

Exercise 3.7

As the agent (robot) is only rewarded for escaping the maze, it does not learn to escape quickly, e.g. the reward for escaping in 10 time steps is the same as escaping in 100 time steps. As such, the agent should be given a reward of -1 at each time step that it does not escape to force it into escaping as quickly as possible.

Exercise 3.8

It is given that $\gamma = 0.5, T = 5$ and $R_1 = -1, R_2 = 2, R_3 = 6, R_4 = 3, R_5 = 2$.
Per definition, $G_T = G_5 = 0$, which gives us

$$
\begin{aligned}
G_5 &= 0 \\
G_4 &= R_5 + 0.5G_5 = 2 + 0 = 2 \\
G_3 &= R_4 + 0.5G_4 = 3 + 1 = 4 \\
G_2 &= R_3 + 0.5G_3 = 6 + 2 = 8 \\
G_1 &= R_2 + 0.5G_2 = 2 + 4 = 6 \\
G_0 &= R_1 + 0.5G_1 = -1 + 3 = 2
\end{aligned}
$$

by means of dynamic programming.

Exercise 3.9

Now, $\gamma = 0.9$ and the reward sequence is $R_1 = 2$ plus an infinite sequence of 7s. Then,

$$G_1 = \frac{7}{1-\gamma} = 70 \tag{1}$$

$$G_0 = R_1 + \gamma G_1 = 2 + 0.9\ G_1 = 65 \tag{2}$$

by equation (3.10) from the book.

Exercise 3.11

We know from literature that a policy π can be defined as $\pi(a|s) = \pi(A_t = a|S_t = s)$ and that the expected rewards for a state-action pair is

$$\begin{aligned} r(s,a) &= \mathbb{E}[R_t|S_{t-1} = s, A_{t-1} = a] \\ &= \sum_{r\in R} r \sum_{s'\in S} p(s', r|s, a) \end{aligned}$$

which gives us the expected reward r given state s and policy π:

$$r(s) = \sum_{a\in A(s)} \pi(a|s) \sum_{r\in R} r \sum_{s'\in S} p(s', r|s, a)$$

where the summation over all possible actions becomes a weighted sum of actions summing up to 1.

Exercise 3.12

By (3.13) and (3.14) from the book it is given that

$$\begin{aligned} v_\pi(s) &= \mathbb{E}[G_t|S_t = s] \\ &= \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma v_\pi(s')], \text{ for all } s \in S \\ &= \sum_a \pi(a|s)\ q_\pi(s, a) \end{aligned}$$

which expresses that the value function equals the summation for all actions of the value of each action in a state times the probability of that action being taken in the given state.

Exercise 3.13

We know from exercise 3.12 that $v_n(s) = \sum_a \pi(a|s)\ q_\pi(s, a)$, and from the definition of the value function $v_n(s)$ in equation (3.12) of the book we know that

$$v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s]$$

which we know from exercise 3.8 can be expressed recursively such that

$$v_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1}|S_t = s]$$

In exercise 3.11 we derived the expected reward given state s and policy π, which inserted into the equation for $v_\pi(s)$ gives us

$$v_\pi(s) = \sum_{a\in A(s)} \pi(a|s) \sum_{r\in R,\ s'\in S} r\ p(s', r|s, a)$$

and from this it can be seen that $v_\pi(s) = \sum_{a\in A(s)} \pi(a|s) + q_\pi(s, a)$, which means that $q_\pi(s, a) = \sum_{r\in R,\ s'\in S} r\ p(s', r|s, a)$.

Exercise 3.14

In the scenario presented in the exercise, the numeral answer is expressed by

$$v_\pi(s_{center}) = \pi(north|s_{center})\ p(s_{north}, r_{north}|s_{center}, a)[r_{north} + \gamma v_\pi(s_{north}] + ...$$

for all of the four future states $s_{north}, s_{south}, s_{west}, s_{east}$.

$$\begin{aligned} v_\pi(s_{center}) &= (0 + 0.9 \cdot 2.3)/4 + \\ &= (0 + 0.9 \cdot (-0.4))/4 + \\ &= (0 + 0.9 \cdot 0.4)/4 + \\ &= (0 + 0.9 \cdot 0.7)/4 \\ &= 0.7 \end{aligned}$$

Exercise 3.15

The sum of infinite terms in equation (3.8) and (3.11) is used:

$$\begin{aligned} v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s] &= \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} + c \mid S_t = s\right] \\ &= \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} + \sum_{k=0}^{\infty} \gamma^k c \mid S_t = s\right] \\ &= \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid S_t = s\right] + \frac{c}{1-\gamma} \end{aligned}$$

so it can be concluded that adding a constant c to all rewards simply adds a constant v_c to all states.

Exercise 3.16

In an episodic task, adding a constant c to all rewards will yield the same result as in exercise 3.15, except for the terminal state G_T. So if we only consider states $s \in S$, we get to the same conclusion as in 3.15.

Exercise 3.17

Based on the Bellman equation for the value function v_π, it can be seen that the Bellman equation for the action value is the same, except for the summation over available actions as the action value function explicitly states an action to be taken (as shown in exercise 3.13). This gives us:

$$q_\pi(s, a) = \sum_{s', r} p(s', r \mid s, a)\left[r + \gamma v_\pi(s')\right], \text{ for all } s \in S$$

Exercise 3.18

Given that we are in state s and select an action a according to some policy π, we have an expectation of the return based on those two parameters. As action a is taken in state s with a probability determined by our policy (as shown in the backup diagram), we get the following equation for the value function:

$$v_\pi(s) = \sum_a \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a]\ \pi(s, a)$$

Substituting the expectation by the action value q_π we get:

$$v_\pi(s) = \sum_a \pi(a|s)\ q_\pi(s, a)$$

which is the value function expressed by π and q_π.