7 Markov Decision Processes

Exercise 7.1

Tell if the following statements about MDPs are true or false. Motivate your answers.

1. To solve an MDP we should take into account state-action pairs one by one;
2. An action you take on an MDP might influence the future rewards you gained;
3. A state considered by the agent acting on an MDP is always equal to the environment state;
4. Problems in which an agent knows the state of the environment and the MDP do not require the use of RL;
5. Policies applied to an MDP are influenced by other learning processes ongoing on the considered MDP.

Exercise 7.2

Tell whether the following statements about MDPs are true or false. Motivate your answers.

1. The Policy Evaluation procedure always outputs the optimal value function.
2. The value function may decrease on some steps of Policy Iteration, but in the end, the algorithm outputs the optimal one.
3. Employing a discount factor in the computation of the cumulative return in MDPs is only a mathematical trick to ensure the convergence of the return.
4. Given an MDP with a certain reward function, there is only a single policy that is optimal for it, and, for each optimal policy, there is only a single reward function for which it is optimal.
5. As many policies can be optimal, there can be multiple optimal value functions in an MDP.
6. All the sequential decision problems can be modeled as MDPs.

Exercise 7.3

State if the following applications may be modeled by means of an MDP:

1. Robotic navigation in a grid world;
2. Stock Investment;
3. Robotic soccer;
4. Playing Carcassonne (board game).

Define the possible actions and states of each MDP you considered.

Exercise 7.4

Consider the following modeling of a classification problem as sequential decision making problem:

$$o_i \leftarrow x_i$$
$$a_i \leftarrow \hat{y}_i$$
$$r_i \leftarrow 1 - |t_i - \hat{y}_i|$$

Does this correspondence makes sense? Comment adequately your answer.

Exercise 7.5

For each one of the following dichotomies in MDP modeling provide examples of problems with the listed characteristics:

1. Finite/infinite actions;
2. Deterministic/stochastic transitions;
3. Deterministic/stochastic rewards;
4. Finite/indefinite/infinite horizon;
5. Stationary/non–stationary environment.

Exercise 7.6

Are the following statements about the discount factor γ in a MDP correct?

- A myopic learner corresponds to have low γ values in the definition of the MDP;

- In an infinite horizon MDP we should avoid using $\gamma = 1$, while it is reasonable if the horizon is finite;
- γ is an hyper-parameter for the policy learning algorithm;
- Probability that an MDP will be played in the next round is γ.

Provide adequate motivations for your answers.

Exercise 7.7

The generic definition of policy is a stochastic function $\pi(h_i) = \mathbb{P}(a_i|h_i)$ which given a history $h_i = \{o_1, a_1, s_1, \ldots, o_i, a_i, s_i\}$ provides a distribution over the possible actions $\{a_i\}_i$.

Formulate the specific definition of a policy if the considered problem is:

1. Markovian, Stochastic, Non-stationary
2. History based, Stochastic, Stationary
3. Markovian, Deterministic, Stationary

Exercise 7.8

Comment the following statements about solving MDPs. Motivate your answers.

1. In a finite state MDP we just need to look for Markovian, stationary and deterministic optimal policies;
2. For finite time MDPs we should consider non-stationary optimal policies;
3. The results of coupling a specific policy and an MDP is a Markov process;
4. Given a policy we can compute P^π and R^π on an MDP;
5. The value function $V^{\pi^*}(s)$ contains all the information to execute the optimal policy π^* on a given MDP;
6. The action-value function $Q^{\pi^*}(s, a)$ contains all the information to execute the optimal policy π^* on a given MDP;
7. There is a unique optimal policy in an MDP;
8. There is a unique optimal value function in an MDP.

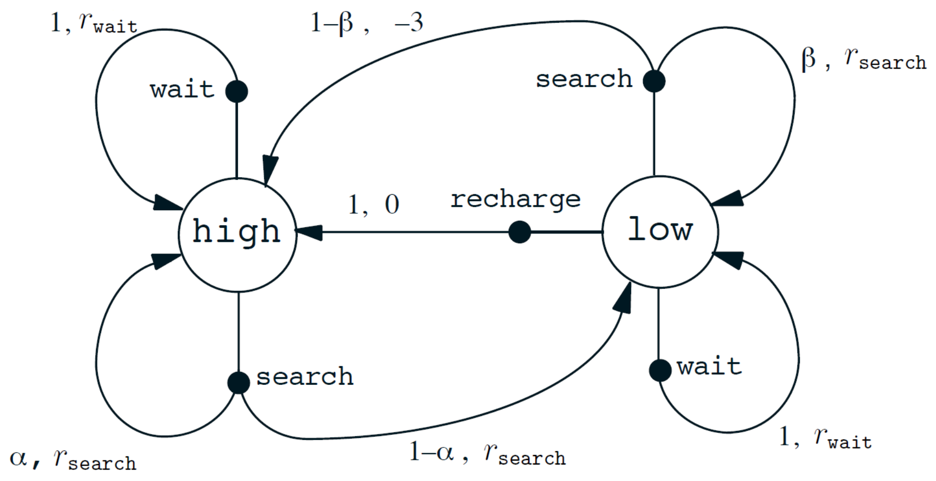

Figure 7.1: The MDP for the cleaning robot problem.

Exercise 7.9

Consider the MDP in Figure 7.1 with $\alpha = 0.3$, $\beta = 0.5$, $\gamma = 1$, $r_{search} = 2$, $r_{wait} = 0$ and the following policy:

$$\pi(s|H) = 1$$
$$\pi(s|L) = 0.5$$
$$\pi(r|L) = 0.5$$

Compute the Value function where the MDP stops after two steps. What happens if we consider a discount factor of $\gamma = 0.5$.

Compute the action-value function for each action value pair in the case the MDP stops after a single step.

Exercise 7.10

Provide the formulation of the Bellman expectation for V equations for the MDP in Figure 7.1, with $\alpha = 0.2$, $\beta = 0.1$, $r_{search} = 2$, $r_{wait} = 0$, $\gamma = 0.9$ and in the case we consider the policy:

$$\pi(H|s) = 1$$
$$\pi(L|r) = 1$$

Exercise 7.11

Tell if the following statements are TRUE or FALSE. Motivate your answers.

1. We are assured to converge to a solution when we apply repeatedly the Bellman expectation operator;
2. We are assured to converge to a solution when we apply repeatedly the Bellman optimality operator;
3. The Bellman solution to Bellman expectation operator is always a good choice to compute the value function for an MDP;
4. The solution provided by the iterative use of the Bellman expectation operator is always less expensive than computing the exact solution using the Bellman expectation equation;
5. The application of the Bellman optimality operator 10 times applied to a generic value function V_0 guarantees that $||V^* - T^{10}V_0||_\infty \leq \gamma^{10}||V^* - V_0||_\infty$

Exercise 7.12

Which one would you chose between the use of the Bellman recursive equation vs. Bellman exact solution in the case we are considering the following problems:

1. Chess
2. Cleaning robot problem in Figure 7.1
3. Maze escape
4. Tic-tac-toe

Provide adequate motivations for your answers.

Exercise 7.13

Consider the MDP in Figure 7.2:

1. Provide the transition matrix for the policy $\pi(I|s_1) = 1, \pi(M|s_2) = 1, \pi(M|s_3) = 1$;
2. Provide the expected instantaneous reward for the previous policy;
3. Compute the value function for the previous policy in the case the MDP stops after two steps;

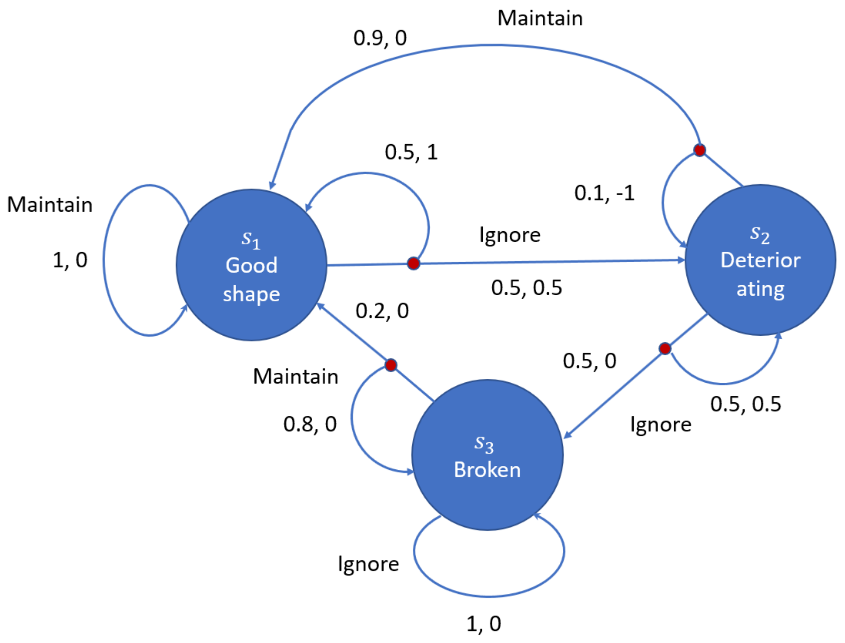

Figure 7.2: The MDP for machinery maintenance problem.

4. Compute the action-value function for each state-action pair in the case the MDP stops after a single step.

Solutions

Answer of exercise 7.1

1. FALSE State/action pairs determines the next state to be visited and the corresponding reward. Thus, to find a proper policy we should consider also the actions chosen in other states.
2. TRUE The action might determine the sequence of states you will visit in the future, thus the reward you will collect.
3. FALSE It depends on the problem we are tackling. Sometimes the state of the environment is completely known to an agent, sometimes it is partially/totally hidden from her. In some problems, even if the environment state is fully known, the agent might consider a different representation of the state to use, for instance if is too complex to be stored.
4. FALSE RL could be used for computational reasons, e.g., if the number of states is too large.
5. TRUE In the case a learner, other than the agent, is operating in the considered MDP we could have a non-stationary MDP.

Answer of exercise 7.2

1. FALSE: Policy Evaluation goal is to find the correct value function corresponding to the policy we are evaluating. The optimal value function is the value function corresponding to the optimal policy instead: In order to obtain it, we have to solve the MDP (e.g., with Value-Iteration).
2. FALSE, the policy improvement theorem guarantees that, at every step, the new policy is better than the previous one.
3. FALSE, the discount factor can also be interpreted as the probability for an episode to terminate at each step, or as how much we value immediate rewards w.r.t. future ones.
4. FALSE, for each MDP we are guaranteed that there is always at least one optimal policy, but this does not prevent having more optimal policies. Moreover, the same policy can still be optimal if we modify the reward function, sometimes this procedure can also speed up learning (reward shaping).
5. FALSE, the only optimal value function is the unique fixed point of the Bellman Optimal Operator, and all optimal policies share the same value function.

6. FALSE, to model a problem as an MDP, we have to assume that the environment state is fully observable and Markovian, hence the current state of the environment should be completely determined by the current observation made by the agent. In some cases, we can "transform" the state of the environment to make it Markovian.

Answer of exercise 7.3

An MDP is fully defined if you specify:

- $\mathcal{S}$ a set of states
- $\mathcal{A}$ a set of actions
- P a state transition probability matrix
- R a reward function, $R(s;a) = \mathbb{E}[r|s;a]$
- γ a discount factor,
- μ_i^0 a set of initial probabilities

Robotic navigation in a grid world:

- $\mathcal{S}$ each one of the locations of the grid
- $\mathcal{A}$ usually is go left, go right, go up and go down, no action in the goal state
- P a binary matrix where you have 1 if you reach s' starting from s with action a and 0 otherwise
- R 0 if you are still away from the goal point and 1 if you reached the goal
- $\gamma = 1$
- μ_i^0 1 in the initial location, 0 otherwise

Stock Investment decision:

- $\mathcal{S}$ amount of money in the bank account, amount of stocks owned
- $\mathcal{A}$ amount of stocks to buy or sell at the next time instant
- P deterministic from a state to another
- R difference in the portfolio value between two time instant
- $\gamma = 0.9$

- μ_i^0 1 in the state corresponding to the amount of money one has in the bank account and the number of stocks she owns, 0 in all other states

Robotic soccer (let us consider the opponent team as a stochastic event):

- $\mathcal{S}$ position of each one of the robots of our team on the field, position of the ball
- $\mathcal{A}$ movement of each one of the robots of the team in the field
- P stochastic transition given from the not deterministic behaviour of the ball and the actions of the environment
- R 1 if a goal has been scored, 0 otherwise
- $\gamma = 1$ since the game ends after a finite amount of time
- μ_i^0 1 in the state corresponding to the initial disposition of the robots on the field and the initial possession of the ball.

Complete rules to play Carcassonne can be found at: `http://riograndegames.com/getFile.php?id=670`. Playing Carcassonne (given the fixed strategies of the other players):

- $\mathcal{S}$ all the possible disposition of the tiles on the table and of the meeples of each player on the tiles in a valid position and a score for each player
- $\mathcal{A}$ place a tile adjacent to a valid tile
- P deterministic state transition from a state to another
- R points scored in the turn or points scored in the final turn
- $\gamma = 1$ since the number of tiles is finite
- μ_i^0 1 on the initial tile on the board, 0 otherwise

Answer of exercise 7.4

The correspondence makes sense, since it is a specific case of an MDP having a single state. The use of traditional techniques to solve MDPs does not make sense, since they are unnecessarily complex for the problem they are trying to solve, i.e, there does not exist a temporal dependence over the predictions.

Answer of exercise 7.5

1. - Finite: robotic navigation (up, down, left and right)

- Infinite actions: pole balancing with continuous space applied force

2. - Deterministic transitions: chess (given the opponent strategy)
   - Stochastic transitions: blackjack
3. - Deterministic rewards: robotic navigation (0 everywhere, 1 in the exit point)
   - Stochastic rewards: ad banner allocation (depending on clicks)
4. - Finite horizon: Carcassonne (finite number of tiles)
   - Indefinite horizon: chess
   - Infinite horizon: stock exchange
5. - Stationary environment: robotic navigation
   - Non-stationary environment: every game with another learner playing

Answer of exercise 7.6

- TRUE Low values of gamma means that we are not considering valuable the revenues gained in the far future. Conversely, in the case we are far–sighted learners, we should use a high value for γ;
- TRUE If we consider $\gamma = 1$ we are not discounting rewards gained in the future, which may lead to produce infinite cumulated rewards. On the contrary, it is reasonable not to discount rewards if we know the horizon is finite;
- FALSE γ is a parameter of the MDP we are considering and it is conceptually related to the problem we are facing and not to the learner. Different values for γ may correspond to different optimal policies for the MDP;
- TRUE By considering a specific $\gamma < 1$ we are somehow stating the fact that the process might end with a given probability at the next time step.

Answer of exercise 7.7

1. Since it is Markovian, the policy will depend only on the current state, i.e., $h_i = s_i$, but will still require information about the time instant we are in since the problem is non-stationary, thus:

$$\pi(s_i, i) = \mathbb{P}(a_i | s_i, i)$$

2. Since the policy is stationary it will not depend on the time instant we are at:

$$\pi(h_i) = \mathbb{P}(a_i|h_i)$$

3. The Markov property induce that we consider only $h_i = s_i$ and the fact that it is deterministic implies that the policy determines a single action at each state:

$$\pi(s_i) = a_i$$

Answer of exercise 7.8

1. TRUE We have insurance that at least one Markovian, deterministic and stationary optimal policy exists. This does not imply that it is only one or that all the optimal policies satisfies the aforementioned properties;
2. TRUE The fact that we know that the process will end implies that the optimal action might be influenced also by the number of rounds remaining;
3. TRUE Once you fix the policy there is no need of deciding anything else and the succession of the states becomes a Markovian process;
4. TRUE Once we fix the policy, we can compute the transition P^π and the reward R^π in each state. This is needed if we want to solve the Bellman expectation equation;
5. FALSE The knowledge of the optimal value in each state $V^{\pi^*}(s)$ does not imply that we know the optimal action to perform in each state to get it;
6. TRUE The state-value function $Q^{\pi^*}(s,a)$ specifies the value obtained at each state for each action. Thus the optimal policy is just $\pi^*(s) = \arg\max_a Q^{\pi^*}(s,a)$;
7. FALSE There might be multiple policies getting the same value in each state;
8. TRUE V^* is the unique fixed point for the Bellman optimality equation.

Answer of exercise 7.9

Since $V^\pi(s) = \sum_{a\in\mathcal{A}} \pi(s|a)\left(R(s,a) + \gamma \sum_{s'\in\mathcal{S}} P(s'|a,s)V^\pi(s)\right)$ thus:

$$\begin{aligned}
V^\pi(H) &= 1[0.3(2+V^\pi(H)) + 0.7(2+V^\pi(L))] \\
&= 0.6 + 0.3(0.3\cdot 2 + 0.7\cdot 2) + 1.4 + 0.7[0.5\cdot 0 + 0.5(0.5\cdot(-3) + 0.5\cdot 2)] \\
&= 0.6 + 0.6 + 1.4 - 0.175 = 2.425 \\
V^\pi(L) &= 0.5(0+V^\pi(H)) + 0.5[0.5(-3+V^\pi(H)) + 0.5(2+V^\pi(L))] \\
&= 0 + 0.5(0.3\cdot 2 + 0.7\cdot 2) - 0.25 + 0.25(0.3\cdot 2 + 0.7\cdot 2) + \\
&+ 0.25[0.5\cdot 0 + 0.5(0.5\cdot(-3) + 0.5\cdot 2)] \\
&= 0 + 1 - 0.25 + 0.5 - 0.0625 = 1.1875
\end{aligned}$$

In the case we have a discount factor $\gamma = 0.5$ we have:

$$\begin{aligned}
V^\pi(H) &= 1[0.3\cdot 2 + 0.5\cdot 0.3V^\pi(H)) + 0.7\cdot 2 + 0.5\cdot 0.7V^\pi(L)] \\
&= 0.6 + 0.15[0.3\cdot 2 + 0.7\cdot 2] + 1.4 + 0.35[0.5\cdot 0 + 0.5(0.5\cdot(-3) + 0.5\cdot 2)]; \\
&= 0.6 + 0.3 + 1.4 - 0.0875 = 2.2125 \\
V^\pi(L) &= 0.5(0 + 0.5V^\pi(H)) + 0.5[0.5(-3 + 0.5V^\pi(H)) + 0.5(2 + 0.5V^\pi(L))] \\
&= 0 + 0.5\cdot 0.5(0.3\cdot 2 + 0.7\cdot 2) - 0.75 + 0.125(0.3\cdot 2 + 0.7\cdot 2) + \\
&+ 0.5 + 0.25[0.5\cdot 0 + 0.5(0.5\cdot(-3) + 0.5\cdot 2)] \\
&= 0.5 - 0.75 + 0.25 + 0.5 - 0.0625 = 0.4375
\end{aligned}$$

Clearly the discounted value for the states is lower than the not discounted one, in the case we consider an MDP with finite time horizon.

The action-value function in our case correspond to the values of the expected instantaneous reward, thus:

$$\begin{aligned}
Q(L,w) &= 1\cdot 0 = 0 \\
Q(L,s) &= 0.5\cdot 2 + 0.5\cdot(-3) = -0.5 \\
Q(L,r) &= 1\cdot 0 = 0 \\
Q(H,w) &= 1\cdot 0 = 0 \\
Q(H,s) &= 0.3\cdot 2 + 0.7\cdot 2 = 2
\end{aligned}$$

Answer of exercise 7.10

$$\begin{aligned}
V(H) &= 2 + 0.9[0.2V(H) + 0.8V(L)] \\
V(L) &= 0 + 0.9V(H)
\end{aligned}$$

Or in matrix form:

$$V = \begin{bmatrix} 2 \\ 0 \end{bmatrix} + 0.9 \begin{bmatrix} 0.2 & 0.8 \\ 1 & 0 \end{bmatrix} V$$

Answer of exercise 7.11

1. TRUE Since it is possible to show that it is a contraction operator, it is assured to have a single optimal point;
2. TRUE As its expectation counterpart it is a contraction, thus converges to an optimal point;
3. FALSE In the case we have computational constraints (or equivalently a huge state space) we might resort to the recursive solution;
4. FALSE If we want to find the exact solution we might need more computational power than solving the linear system corresponding to the Bellman expectation equation;
5. TRUE Since the fact that the operator is a contraction will guarantee to shrink the distance from the optimal solution at each step of a factor γ.

Answer of exercise 7.12

1. RECURSIVE The state space is too large
2. EXACT The state space is small enough that the exact solution is feasible
3. NONE We do not have information about the state we are in
4. EXACT Again we have a small enough state space

Answer of exercise 7.13

1.
$$P = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0.9 & 0.1 & 0 \\ 0.2 & 0 & 0.8 \end{bmatrix}$$

2.
$$R = \begin{bmatrix} 0.75 \\ -0.1 \\ 0 \end{bmatrix}$$

3.

$$\begin{aligned}
V(s_1) &= R(s_1) + 0.5V(s_1) + 0.5V(s_2) \\
&= R(s_1) + 0.5R(s_1) + 0.5R(s_2) = 0.75 - 0.1 + 0.75 = 1.4 \\
V(s_2) &= R(s_2) + 0.9V(s_1) + 0.1V(s_2) \\
&= R(s_2) + 0.9R(s_1) + 0.1R(s_2) = -0.1 + 0.675 - 0.01 = 0.664 \\
V(s_3) &= R(s_3) + 0.8V(s_1) + 0.2V(s_3) \\
&= R(s_3) + 0.8R(s_1) + 0.2R(s_3) = 0 + 0.15 + 0 = 0.15
\end{aligned}$$

4.

$$\begin{aligned}
&Q(M, s_1) = 0 \\
&Q(I, s_1) = 0.5 \cdot 0.5 + 1 \cdot 0.5 = 0.75 \\
&Q(M, s_2) = -1 \cdot 0.1 + 0 \cdot 0.9 = -0.1 \\
&Q(I, s_2) = 0 \cdot 0.5 + 0.5 \cdot 0.5 = 0.25 \\
&Q(M, s_3) = 0 \cdot 0.8 + 0 \cdot 0.2 = 0 \\
&Q(I, s_3) = 0
\end{aligned}$$