Prof. Joschka Bödecker, Gabriel Kalweit

Reinforcement Learning
Sample Solution 1

1 Markov Decision Processes

In this exercise, we will deal with finite horizon problems of length N. Consider a student $\mathcal{S}$ who takes a written exam. The exam consists of k different tasks to be solved, each will provide r_i ($i \in \{1, \dots, k\}$, see Table 1) points. To keep the problem easy, we assume that each task can be solved only either completely wrong or completely correct (in the first case 0 points, in the latter r_i) and that the student knows with absolute certainty whether a task was solved properly or not after working on it. Hence, the student acts deterministically.
The student $\mathcal{S}$ can try to solve each task arbitrarily often. Each try will be successful with a probability $p_i^{\mathcal{S}}$ which depends on the difficulty of the task and the knowledge of the student (these probabilities are listed in Table 1).

(a) Formalize the above described problem as a Markov Decision Process. *Hint: It can make sense to also include the next state in the reward function.*

Solution.

- MDP: $M = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$
- States: $\mathcal{S} = \{i = (i_1, \dots, i_k) | i_v \in 0, 1\}$.
  - Every task can be either solved or unsolved. Therefore we have 2^k possible states. Therefore $|\mathcal{S}| = 2^3 = 8$.
- Actions: The student can try to solve an unsolved task, i.e. the set of actions varies over time and represents the current set of unsolved tasks. $\mathcal{A}(s) = \{i | s_i = 0\}$
- The transition function is stochastic and depends on the previously given probability table.
  - For example: $p_{ij}(u) = 0.1$ for $i = (0, 0, 1)$ and $j = (1, 0, 1)$ if $u = u_1$.
- The reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A}$ has to correspond to the optimization goal, this means in our case, to achieve as many points in the exam as possible.
  - In the current scenario, the rewards depend not only on the current state and the action, but also on the following state. This is in the literature commonly modeled as $\mathcal{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S}$.
  - We set
$$\mathcal{R}(i, a, j) = \begin{cases} 0 & i = j \\ r_a & \text{else} \end{cases}$$
where r_a denotes the number of points that are given for solving task a. (see Table).

- Since this is a finite horizon problem of length 3, we set $\gamma = 1$.

Of course, there is a time limit in exams. We assume that the duration of the exam allows the student a total of up to $N = 3$ tries to solve the tasks (again to simplify, we assume each try takes the same time). It is also known that to pass the exam at least 50 % of the maximum score must be achieved.

(b) How would you model the risk of failing the exam in this scenario? *Hint: You can introduce terminal rewards, i.e. rewards for getting in some terminal state that causes the episode to end.*

Solution.

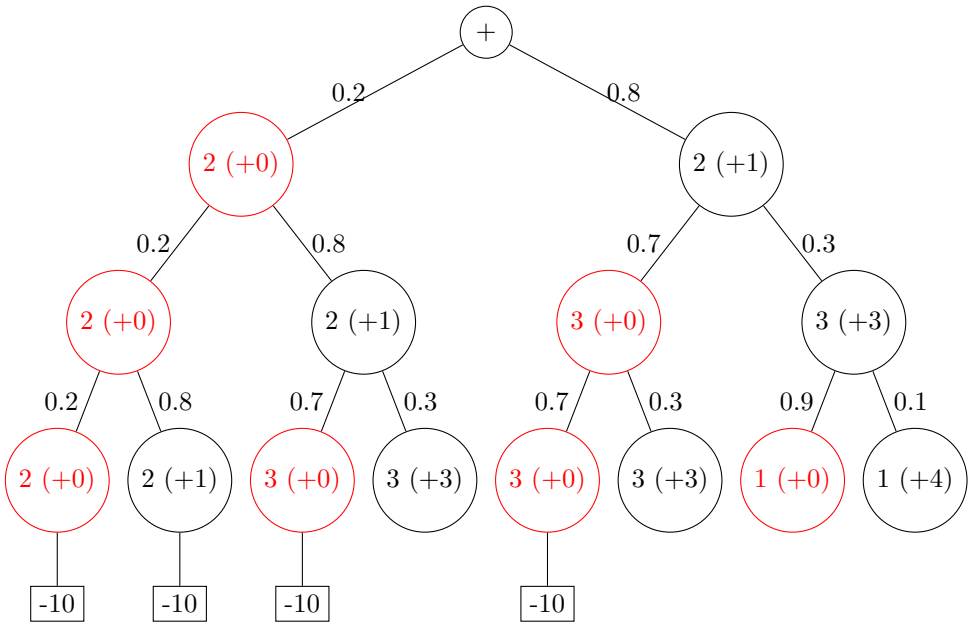
- In this scenario, the student needs at least 4 of 8 points to pass the exam.
- In this N-step decision problem every state has terminal rewards $g(i)$, which arise when the system terminates after N steps in state i.
- The number of achieved points (and therefore the relation to the grade – higher number of collected points mean higher rewards and a higher grade) we have already modeled with the direct rewards.
- For terminal states i, in which the exam is passed, we not necessarily need to define a terminal reward, i.e. $g(i) = 0$.
- For terminal states i, in which we only collected 3 or less points, we define $g(i) = -C$.
- To define the relation of $-C$ to the direct rewards we leave the student to decide.
  - For example, if to pass the exam is obligatory, the student would choose C to be very high.
- In the following we set $C = 10$.

In the following, we will compare several policies to solve the exam, i.e. which unsolved exercise a student chooses to solve next. In case of a failed exam, for all subsequent calculations, we assume terminal rewards of -10 for failing states.

(c) Student $\mathcal{S}$ considers two possible policies, $\pi_A^{\mathcal{S}}$ and $\pi_B^{\mathcal{S}}$, that determine in which order the tasks will be solved. The first policy tackles the tasks in increasing difficulty, i.e. with decreasing possibility of success $p_i^{\mathcal{S}}$. Following policy $\pi_B^{\mathcal{S}}$, the tasks are solved in reversed order (reversed with respect to $\pi_A^{\mathcal{S}}$). The student knows about the success of solving a task after dealing with it and sticks with a task until it is solved correctly. Compare both policies by determining the values for following $\pi_A^{\mathcal{S}}$ and $\pi_B^{\mathcal{S}}$ starting in the initial state of not having solved a task yet.

Solution. $v_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s]$. In our case, the policy is deterministic hence becomes a conditional Dirac delta function $P[a|s] = \delta_{a=\lambda(s)}$ for some action selector λ. Furthermore, the reward function depends on s' as well. Lastly, we have a finite horizon problem of length 3. When there is no next state, we get the terminal rewards.

- Policy A

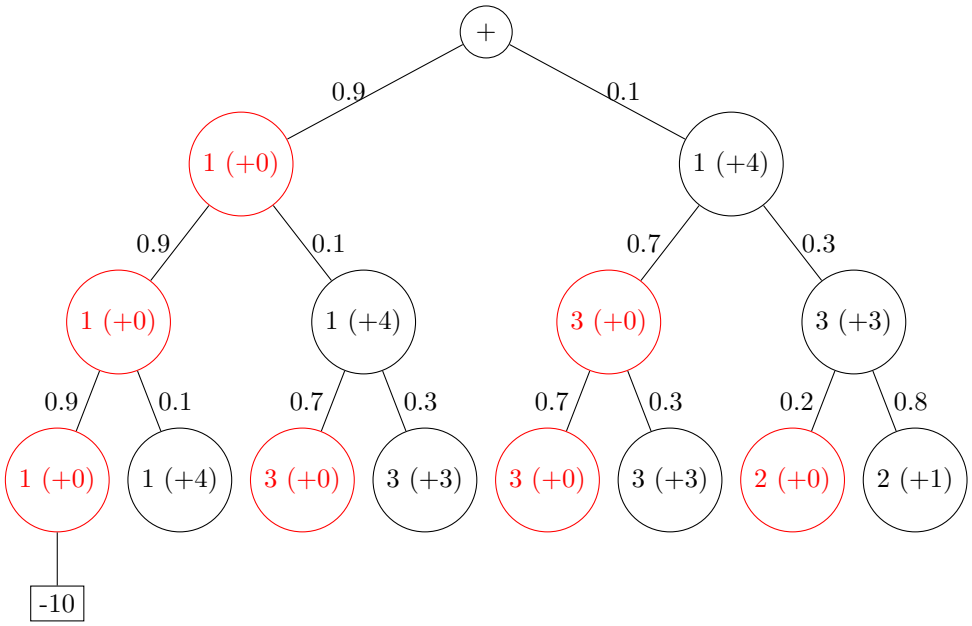


Order of chosen actions: 2,3,1	Actions
$v_{\pi_A^S}(s) = (1-0.8)^3 \cdot (-10) +$	2, 2, 2
$(1-0.8)^2 \cdot 0.8 \cdot (-10+1.0) +$	2, 2, 2
$(1-0.8) \cdot 0.8 \cdot (1-0.3) \cdot (-10+1.0) +$	2, 2, 3
$(1-0.8) \cdot 0.8 \cdot 0.3 \cdot (1.0+3.0) +$	2, 2, 3
$0.8 \cdot (1-0.3) \cdot (1-0.3) \cdot (-10+1.0) +$	2, 3, 3
$0.8 \cdot (1-0.3) \cdot 0.3 \cdot (1.0+3.0) +$	2, 3, 3
$0.8 \cdot 0.3 \cdot (1-0.1) \cdot (1.0+3.0) +$	2, 3, 1
$0.8 \cdot 0.3 \cdot 0.1 \cdot (1.0+3.0+4.0) = -2.984$	2, 3, 1

Task	Points	Probability of Success p_i^S
u_1	4	0.1
u_2	1	0.8
u_3	3	0.3

Table 1: Properties of exam tasks.

- Policy B



	Order of chosen actions: 1,3,2	Actions
$v_{\pi_B^S}(s) =$	$(1-0.1)^3 \cdot (-10) +$	1, 1, 1
	$(1-0.1)^2 \cdot 0.1 \cdot (4.0) +$	1, 1, 1
	$(1-0.1) \cdot 0.1 \cdot (1-0.3) \cdot (4.0) +$	1, 1, 3
	$(1-0.1) \cdot 0.1 \cdot 0.3 \cdot (4.0+3.0) +$	1, 1, 3
	$0.1 \cdot (1-0.3) \cdot (1-0.3) \cdot (4.0) +$	1, 3, 3
	$0.1 \cdot (1-0.3) \cdot 0.3 \cdot (4.0+3.0) +$	1, 3, 3
	$0.1 \cdot 0.3 \cdot (1-0.8) \cdot (4.0+3.0) +$	1, 3, 2
	$0.1 \cdot 0.3 \cdot 0.8 \cdot (4.0+3.0+1.0) = -5.948$	1, 3, 2

$v_{\pi_A^S} = -2.984$ and $v_{\pi_B^S} = -5.948$, thus Strategy A is better.

2 Markov Property

Imagine you want to apply the algorithms from this lecture on a real physical system. You get sensor input after each 0.01 seconds, but the execution of actions has a delay of 0.2 seconds.

(a) Is the Markov property fulfilled?

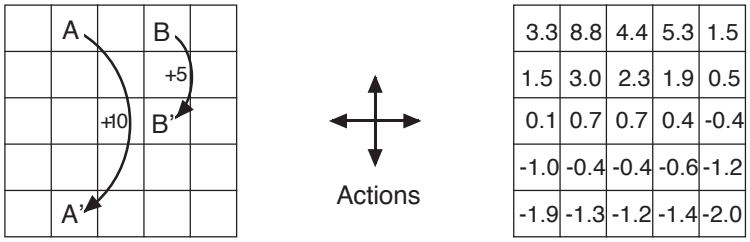
Solution. The action chosen by the agent in a current time step is not the action which is executed, but one that got chosen 0.2 seconds ago. The next state depends on the history of actions. The Markov property is not fulfilled.

(b) Is the Markov property fulfilled if you consider the history of the last 0.2 seconds as part of the state space?

Solution. The action that leads to a certain next state is part of the history and therefore now part of the state space. The Markov property is fulfilled. However, this blows up the state space and directly leads us to the curse of dimensionalty.

3 Bellman Equation

In the grid world shown in Figure 1, the cells of the grid correspond to states. In each cell, four actions can be taken (north, south, east and west) which move the agent deterministically in the respective neighboring cell. Actions that would move the agent out of the grid won't lead to cell changes, but cause direct costs of -1. All other actions are free, with the exception of the states A and B. Every action performed by the agent in A moves it in A' with a reward of 10, each action in B moves it to B' with a reward of 5. Assume that the agent following policy π selects all actions with equal probability in all states. The accompanying value function v_π with a discounting factor of $\gamma = 0.9$ is given in the right part of Figure 1.



3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

Figure 1: gridworld

(a) Show exemplary for state $s_{0,0}$ with $v_\pi(s_{0,0}) = 3.3$ that the Bellman equation is satisfied.

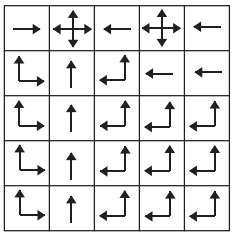


Figure 2: Optimal policy for gridworld.

Solution. Going up or left cause a reward of -1 and the agent stays in $s_{0,0}$.

$$\begin{aligned} v_\pi(s_{0,0}) &= 0.25 \cdot (-1 + \gamma \cdot 3.3) + \\ &\quad 0.25 \cdot (0 + \gamma \cdot 8.8) + \\ &\quad 0.25 \cdot (0 + \gamma \cdot 1.5) + \\ &\quad 0.25 \cdot (-1 + \gamma \cdot 3.3) \\ &= 3.3025 \\ &\approx 3.3 \end{aligned}$$

(b) Explain why the value of state B is higher than the direct reward. Why does this not hold for state A?

Solution.

- Lets first consider state B': It has positive expected reward since the risk of running into a wall from here (costs of -1) is more than compensated by the chance to reach – randomly – state A or B.
- This in contrary does not hold for state A': Here the risk of running into a wall is much higher due to the random policy. Therefore the expected reward $v_\pi(A')$ is lower than zero.
- Now consider state B: It has higher value (+5.3) than direct reward (+5), since the agent jumps in state B always to B' which also has positive value (+0.4).
- Finally we should mention that for the agent using policy π state A is the *best* state overall. However, in contrary to state B its expected value (+8.8) is not as high as its direct reward (+10).

The optimal policy π_* is shown in Figure 2. Now assume that we reduce the reward for jumping from A to A' to 4.

(c) How does the optimal policy change? What are the new values of states A and B?

Solution. The best state obviously becomes now B (higher value for encountering B'). Hence, the optimal policy changes in that all paths now lead to B (instead of A).

$$\begin{aligned} v_*(B) &= 5 + \gamma v_*(B') \\ &= 5 + \gamma(0 + \gamma v_*(s_{1,3})) \\ &= 5 + \gamma(0 + \gamma(0 + \gamma v_*(B))) \\ &= 5 + \gamma^3 v_*(B) \end{aligned}$$

And thus $v_*(B) = \frac{5}{1-\gamma^3} \approx 18.45$ and $v_*(A) = 4 + \gamma v_*(A') = 4 + \gamma^7 v_*(B) \approx 12.82$.