Autonomous Systems

H. Geffner | Universitat Pompeu Fabra
A. Occhipinti (exercise description by G. Francès) | 2021-2022 Term 2

Exercise Sheet D

Due: March 21, 2021

The last assignment of the Autonomous Systems lab will consist on defining some Markov Decision Processes and implementing some of the solution techniques you have seen in class. We will again use the Berkeley Pacman environment for some exercises.

Exercise D.1 (Berkeley Pacman Exercises Q1-4: 5 points)
Follow the exercises at

`https://inst.eecs.berkeley.edu/~cs188/sp21/project6/`

and answer questions 1 to 4. The idea is here is to implement and experiment a bit with the techniques seen in class (e.g. value iteration, Q-learning). By following the instructions on this Pacman exercise, you should be able to easily visualize the size of problems that each of these methods is able to solve.

For this exercise, you are asked to submit all the relevant code for questions Q1 to Q4.

Exercise D.2 (Push Your Luck: 5 points)
This is a modeling-only exercise, no need to code anything here, just to think (!)

Let us hypothetically suppose that after all the hard study and practice (this is the hypothetical part 😄) during the Autonomous Systems course, you want to put all the knowledge you have amassed about MDPs *to good use*. You've heard that the latest trend in the Casino of Barcelona is *Push your Luck*, a simple dice game where the player has the chance of playing a number of *episodes* at the end of which she collects some good money. Each episode consists on the player rolling a fair die a number of times. After each die roll, the player can choose to (a) finish the episode, in which case she collects the *accumulated reward* of that episode, or (b) to roll again. The accumulated reward of an episode is defined as the product of all die outcomes for that episode. There is one glitch though (there always is!): If the die shows a number that has already appeared in the current episode, then the player gets no reward at all. So, for instance if a player rolls a 3, a 4, a 1, and decides to end the episode, her accumulated reward will be $3 \cdot 4 \cdot 1 = 12$. If, on the contrary, she decides to keep on playing (of course hoping to get a 6), and she gets a 4, then the episode finishes with no reward at all, because the 4 had already appeared. Look at some sample episodes for you to get a feeling of the game:

- $\langle 3, 4, 1, STOP \rangle$: Player gets 12. This is the above example, and player decided that a reward of 12 was better than risking losing everything.
- $\langle 6, 2, 1, 6 \rangle$: Player gets nothing, since 6 appears twice.
- $\langle 1, 2, 3, 4, 5, 3 \rangle$: Player gets nothing — she was fooled by the appearence of a beautiful sequence, but probabilities are cruel.
- $\langle 1, 1 \rangle$: Player gets nothing — a very unlucky episode!.
- $\langle 3, 5, 6, STOP \rangle$: Player gets 90 — quite a good episode.
- $\langle 6, 2, 4, 3, 5 \rangle$: Player gets 720 — can't get any better than this!

(a) Provide the formalization of the above problem as an infinite-horizon *discounted reward* MDP

$$\mathcal{M} = \langle S, A, (P_a)_{a \in A}, r, s_0, \gamma \rangle,$$

where the meaning of each element is as seen in class. This means that you need to describe the set S of states for your problem, the initial state s_0, the set A of actions, the transition probabilities $P_a(s'|s)$ of going from state s to state s' when taking action a (for each action a and pair of states s, s'), and the reward function r that encodes the reward $r(a, s)$ obtained when taking action a in state s. To be faithful to the model seen in class, you also need to provide a discount factor γ, but this is not relevant for the correctness of your model, only for the solutions, so you can assume any factor you prefer for now. Note that you here assume that the number of episodes is unbounded, i.e. player keeps playing forever.

(b) Discuss briefly what should be changed in your model in order to work for games with dice having an arbitrary number N of sides.

For this exercise, you are asked to submit written answers to the two questions, including a formal description of each element of the MDP.

The exercise sheets should be submitted in groups of two or three students. Please submit one single copy of the exercises per group (only one member of the group does the submission), and provide all student names on the submission.