TÉCNICO LISBOA

Planning, Learning and Decision Making

MSc in Computer Science and Engineering

Re-sit examination — July 18, 2022

Instructions

- You have 120 minutes to complete the examination.
- Make sure that your test has a total of 9 pages and is not missing any sheets, then write your full name and student n. on this page (and your number in all others).
- The test has a total of 4 questions, with a maximum score of 20 points. The questions have different levels of difficulty. The point value of each question is provided next to the question number.
- *If you get stuck in a question, move on.* You should start with the easier questions to secure those points, before moving on to the harder questions.
- In the multiple choice questions, *you do not get negative points* if you get the answer wrong.
- Please provide your answer in the space below each question, and make sure to include all relevant computations. If you make a mess, clearly indicate your answer.
- The exam is open book and open notes. You may use a calculator, but any other type of electronic or communication equipment is not allowed.
- Good luck.

Question 1. (**2.5 pts.**)

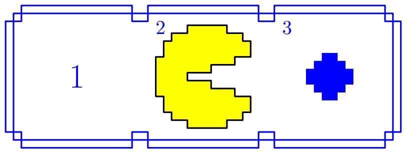


Figure 1: Pacman and ghost in a 1×3 grid.

Consider the environment in Fig. 1, representing the Pacman scenario that you already encountered in several homework assignments. In this problem you will describe the decision problem faced by a player controlling the Pacman character.

The player has 3 actions available: "Left", "Right", and "Stay", the first two of which move Pacman one step in the corresponding direction, if an adjacent cell exists in that direction. Otherwise, Pacman remains in the same place. Pacman can be in any of the 3 numbered cells. Besides Pacman, there is a power pellet in the environment (the blue shape in cell 3 in Fig. 1). The pellet can be in any of the three cells, and Pacman cannot observe in which cell the pellet is.

If Pacman lies in the same cell as the pellet, it "eats" the pellet; the pellet disappears from that cell and reappears in one of the two other cells with equal probability. Pacman's goal is to eat as many pellets as possible.

Describe the decision problem faced by the agent using the adequate type of model. In particular, you should indicate:

- The type of model needed to describe the decision problem of the agent;
- The state, action, and observation space (if relevant);
- The transition probabilities corresponding to the action "Move Left";
- The observation probabilities for the action "Move Left" (if relevant);
- The immediate cost function.

Make sure that

- The cost function is as simple as possible and verifies $c(x, a) \in [0, 1]$ for all states $x \in \mathcal{X}$ and actions $a \in \mathcal{A}$.
- The cost depends only on the state of the environment.

Solution 1.

The decision problem can be modeled as a partially observable Markov decision problem $(\mathcal{X}, \mathcal{A}, \mathcal{Z}, \{\mathbf{P}_a\}, \{\mathbf{O}_a\}, c, \gamma)$, where:

- The state space $\mathcal{X}$ consists of all pairs (p, b), with $p, b \in \{1, 2, 3\}$. The component p indicates the position of Pacman, and b the position of the pellet. In other words, $\mathcal{X} = \{(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3)\}$.
- The action space $\mathcal{A}$ is such that $\mathcal{A} = \{L, R, S\}$, corresponding to the three actions available to the player.
- The observation space is $\mathcal{Z} = \{1, 2, 3\}$, corresponding to the position of Pacman.
- The transition probabilities for the action "Move Left" are given by

$$\mathbf{P}_L = \begin{bmatrix} 0.0 & 0.5 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.5 & 0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.5 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}.$$

- The observation probabilities are action independent, so we have

$$\mathbf{O}_L = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}.$$

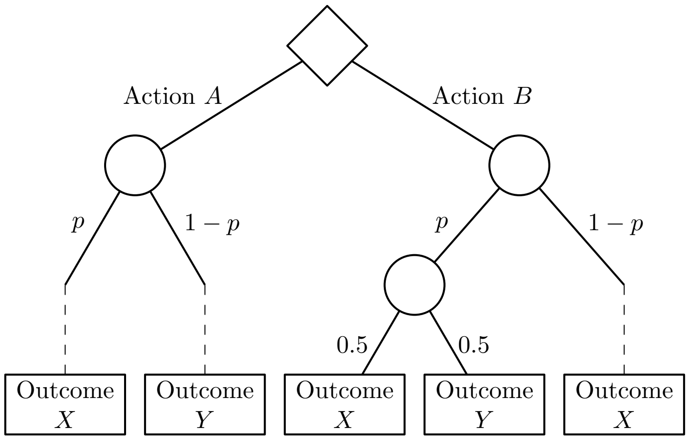
- Finally, the cost is minimal whenever Pacman is in the same position as the pellet, and maximal otherwise, yielding

$$\boldsymbol{C} = \begin{bmatrix} 0.0 & 0.0 & 0.0 \\ 1.0 & 1.0 & 1.0 \\ 1.0 & 1.0 & 1.0 \\ 1.0 & 1.0 & 1.0 \\ 0.0 & 0.0 & 0.0 \\ 1.0 & 1.0 & 1.0 \\ 1.0 & 1.0 & 1.0 \\ 1.0 & 1.0 & 1.0 \\ 0.0 & 0.0 & 0.0 \end{bmatrix}.$$

Question 2. (8 pts.)

For each of the following questions, indicate the *single most correct answer.*

(a) **(0.8 pts.)** Consider the following decision tree:



Suppose that $u(X) = 0$ and $u(Y) = 1$. Which of the following holds?

☐ Action A is preferred to action B if $p > 1/2$.

☒ **Action B is preferred to action A if $p > 2/3$.**

☐ Action B is never preferred to action A.

☐ None of the above.

(b) **(0.8 pts.)** Consider a Markov chain $(\mathcal{X}, \mathbf{P})$, where $\mathcal{X} = \{A, B, C\}$ and

$$\mathbf{P} = \begin{bmatrix} 0.3 & 0.1 & 0.6 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.7 & 0.3 \end{bmatrix}.$$

☐ The Markov chain has a single communicating class, $\{A, B, C\}$.

☐ The communicating classes for the chain are $\{B\}$ and $\{A, C\}$.

☒ **The communicating classes for the chain are $\{A\}$, $\{B\}$, and $\{C\}$.**

☐ The chain is irreducible.

(c) **(0.8 pts.)** A Markov chain $(\mathcal{X}, \mathbf{P})$ is ergodic if...

☐ ... $d(x) = 1$ for all $x \in \mathcal{X}$, where $d(x)$ is the period of x.

☒ **... for any initial distribution μ_0, $\lim_{t\to\infty} \mu_0 \mathbf{P}^t = \mu^*$, where μ^* is the stationary distribution for the chain.**

☐ ... it is irreducible, i.e., for any $x, y \in \mathcal{X}$ there is $t \geq 0$ such that $\mathbf{P}^t(y \mid x) > 0$.

☐ None of the above.

(d) **(0.8 pts.)** Consider an HMM $(\mathcal{X}, \mathcal{Z}, \mathbf{P}, \mathbf{O})$ with $\mathcal{X} = \{A, B, C\}$, $\mathcal{Z} = \{u, v\}$, and

$$\mathbf{P} = \begin{bmatrix} 0.3 & 0.1 & 0.6 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.7 & 0.3 \end{bmatrix}, \qquad \mathbf{O} = \begin{bmatrix} 0.3 & 0.7 \\ 0.7 & 0.3 \\ 0.5 & 0.5 \end{bmatrix}.$$

Suppose that $\boldsymbol{\mu}_0 = \begin{bmatrix} 0.33 & 0.33 & 0.33 \end{bmatrix}$ and the agent observes $\boldsymbol{z}_{0:1} = \{u, u\}$. If we run the forward algorithm on the sequence of observations $\boldsymbol{z}_{0:1}$, then...

☐ $\boldsymbol{\alpha}_0 = \begin{bmatrix} 0.33 & 0.33 & 0.33 \end{bmatrix}$.

☐ $\boldsymbol{\alpha}_1 = \begin{bmatrix} 0.10 & 0.23 & 0.17 \end{bmatrix}$.

☐ $\boldsymbol{\mu}_{1|0:1} = \begin{bmatrix} 0.00 & 0.25 & 0.06 \end{bmatrix}$.

☒ $\boldsymbol{\mu}_{1|0:1} = \begin{bmatrix} 0.03 & 0.80 & 0.17 \end{bmatrix}$**.**

(e) **(0.8 pts.)** Consider once again the HMM from the previous question, and suppose that for some sequence of observations $z_{0:2}$, we run the Viterbi algorithm, to compute the most likely sequence, and get

$$\boldsymbol{i}_1 = \begin{bmatrix} 1 & 3 & 1 \end{bmatrix}, \qquad \boldsymbol{i}_2 = \begin{bmatrix} 1 & 3 & 1 \end{bmatrix}, \qquad \boldsymbol{m}_2 = \begin{bmatrix} 0.00 & 0.03 & 0.02 \end{bmatrix}.$$

The most likely sequence of states given $z_{0:2}$ is

☒ $\boldsymbol{x}_{0:2} = \{A, C, B\}$**.**

☐ $\boldsymbol{x}_{0:2} = \{B, C, A\}$.

☐ $\boldsymbol{x}_{0:2} = \{C, B, B\}$.

☐ None of the above.

(f) **(0.8 pts.)** Inverse reinforcement learning is an *ill-posed problem* because...

☐ ... the learner does not have access to samples of the cost function to be learned.

☒ **... a policy can be optimal for multiple cost functions (in particular, trivial ones).**

☐ ... the learner must recover a full policy given only examples by a teacher.

☐ ... if the learner makes no assumptions, it cannot recover a cost function given only samples from the optimal policy.

(g) **(0.8 pts.)** In *Bayesian IRL*, given a demonstration provided by a teacher, the learner seeks to determine a cost function...

☐ ... that maximizes the likelihood of the demonstration.

☒ **... with maximal posterior probability given the demonstration.**

☐ ... that minimizes the negative log-likelihood of the demonstration given the cost function.

☐ ... that replicates the observed demonstration as closely as possible.

(h) **(0.8 pts.)** In the *weighted majority algorithm*, the maximum number of mistakes is upper bounded by...

☐ ... $\log_2(N)$, where N is the number of sources/experts available to the predictor.

☐ ... $\sqrt{T\log(N)}$, where N is the number of sources/experts available to the predictor and T is the number of time steps.

☐ ... $\sqrt{N\log(T)}$, where N is the number of sources/experts available to the predictor and T is the number of time steps

☒ **None of the above.**

(i) **(0.8 pts.)** The *exponentially weighted averager* algorithm is most suited to address...

☐ ... adversarial multi-armed bandit problems.

☐ ... stochastic multi-armed bandit problems.

☒ **... non-stochastic sequential prediction problems.**

☐ ... the exploration-exploitation problem in reinforcement learning.

(j) **(0.8 pts.)** In POMDPs, *point-based methods* are...

☐ ... a class of exact solution methods based on policy iteration.

☒ **... a class of non-exact solution methods that rely on sampling the belief-space.**

☐ ... a class of non-exact solution methods that make use of the solution to the underlying MDP.

☐ ... a class of exact solution methods that make use of α-vectors.

In the remainder of the test, consider the MDP $\mathcal{M} = (\mathcal{X}, \mathcal{A}, \{\mathbf{P}_a\}, c, \gamma)$ where

- $\mathcal{X} = \{A, B, C\}$;
- $\mathcal{A} = \{a, b, c\}$;
- The transition probabilities are

$$\mathbf{P}_a = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}; \qquad \mathbf{P}_b = \begin{bmatrix} 0.2 & 0.8 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.8 & 0.2 \end{bmatrix}; \qquad \mathbf{P}_c = \begin{bmatrix} 0.2 & 0.8 & 0.0 \\ 0.0 & 0.2 & 0.8 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}.$$

- The cost function c is given by

$$\mathbf{C} = \begin{bmatrix} 0.8 & 1.0 & 1.0 \\ 0.8 & 1.0 & 1.0 \\ 0.0 & 0.2 & 0.2 \end{bmatrix}.$$

- Finally, the discount is given by $\gamma = 0.9$.

Question 3. (4 pts.)

Consider the MDP $\mathcal{M}$ defined in the previous shaded box and suppose that, for some policy π,

$$\boldsymbol{J}^\pi = \begin{bmatrix} 2.29 & 1.22 & 0.0 \end{bmatrix}^\top.$$

(a) **(2.0 pts.)** Compute Q^π. Is the policy π optimal? Explain your reasoning.

(b) **(1.0 pt.)** Consider now an MDP $\mathcal{M}' = (\mathcal{X}, \mathcal{A}, \{P_a\}, c', \gamma)$ obtained from $\mathcal{M}$ by considering a cost function $c' = c + k$, for some constant k. Compute J^π in $\mathcal{M}'$. Is the policy π optimal for $\mathcal{M}'$?

(c) **(1.0 pts.)** Consider another MDP $\mathcal{M}'' = (\mathcal{X}, \mathcal{A}, \{P_a\}, c'', \gamma)$ obtained from $\mathcal{M}$ by now considering a cost function $c'' = c \times k$, for some constant k. Compute J^π in $\mathcal{M}''$. Is the policy π optimal for $\mathcal{M}''$?

Solution 3.

(a) We have that

$$Q^\pi(x, a) = c(x, a) + \gamma \sum_{x' \in \mathcal{X}} \mathbf{P}(x' \mid x, a) J^\pi(x').$$

Performing the above computation columnwise, we get

$$\begin{aligned}
Q_a^\pi &= c_a + \gamma \mathbf{P}_a \boldsymbol{J}^\pi \\
&= \begin{bmatrix} 0.8 \\ 0.8 \\ 0.0 \end{bmatrix} + 0.9 \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix} \begin{bmatrix} 2.29 \\ 1.22 \\ 0.0 \end{bmatrix} = \begin{bmatrix} 2.86 \\ 1.90 \\ 0.0 \end{bmatrix}, \\
Q_b^\pi &= c_b + \gamma \mathbf{P}_b \boldsymbol{J}^\pi \\
&= \begin{bmatrix} 1.0 \\ 1.0 \\ 0.2 \end{bmatrix} + 0.9 \begin{bmatrix} 0.2 & 0.8 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.8 & 0.2 \end{bmatrix} \begin{bmatrix} 2.29 \\ 1.22 \\ 0.0 \end{bmatrix} = \begin{bmatrix} 2.29 \\ 2.10 \\ 1.08 \end{bmatrix}, \\
Q_c^\pi &= c_c + \gamma \mathbf{P}_c \boldsymbol{J}^\pi \\
&= \begin{bmatrix} 1.0 \\ 1.0 \\ 0.2 \end{bmatrix} + 0.9 \begin{bmatrix} 0.2 & 0.8 & 0.0 \\ 0.0 & 0.2 & 0.8 \\ 0.0 & 0.0 & 1.0 \end{bmatrix} \begin{bmatrix} 2.29 \\ 1.22 \\ 0.0 \end{bmatrix} = \begin{bmatrix} 2.29 \\ 1.22 \\ 0.2 \end{bmatrix}.
\end{aligned}$$

The resulting Q-function is, thus,

$$Q^\pi = \begin{bmatrix} 2.86 & 2.29 & 2.29 \\ 1.90 & 2.10 & 1.22 \\ 0.0 & 1.08 & 0.2 \end{bmatrix}.$$

Since we can observe that, for every state $x \in \mathcal{X}$, $J^\pi(x) = \min_a Q(x, a)$, we can conclude that J^π verifies the recursive relation

$$J^\pi(x) = \min_{a \in \mathcal{X}} \left[c(x, a) + \gamma \sum_{x' \in \mathcal{X}} \mathbf{P}(x' \mid x, a) J^\pi(x') \right],$$

which is the recursive relation for J^*, we can conclude that $J^\pi = J^*$ and π is optimal.

(b) We have that, in $\mathcal{M}$,

$$J^\pi(x) = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \mathrm{c}_t \mid \mathrm{x}_0 = x\right],$$

In $\mathcal{M}'$, we have that

$$\begin{aligned}
J^\pi_{\mathcal{M}'}(x) &= \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \mathrm{c}'_t \mid \mathrm{x}_0 = x\right] \\
&= \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t (\mathrm{c}_t + k) \mid \mathrm{x}_0 = x\right] \\
&= \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \mathrm{c}_t \mid \mathrm{x}_0 = x\right] + \sum_{t=0}^{\infty} \gamma^t k \\
&= J^\pi_{\mathcal{M}}(x) + \frac{k}{1-\gamma},
\end{aligned}$$

due to the linearity of the expectation. Note also that

$$\begin{aligned}
Q^\pi_{\mathcal{M}'}(x,a) &= c'(x,a) + \gamma \sum_{x' \in \mathcal{X}} \mathbf{P}(x' \mid x,a) J^\pi_{\mathcal{M}'}(x) \\
&= c(x,a) + k + \gamma \sum_{x' \in \mathcal{X}} \mathbf{P}(x' \mid x,a) J^\pi_{\mathcal{M}}(x) + \frac{\gamma k}{1-\gamma} \\
&= Q^\pi_{\mathcal{M}}(x,a) + \frac{k}{1-\gamma}.
\end{aligned}$$

Therefore, much like in (a),

$$\min_{a \in \mathcal{X}} Q^\pi_{\mathcal{M}'}(x,a) = \min_{a \in \mathcal{X}} \left[Q^\pi_{\mathcal{M}}(x,a) + \frac{k}{1-\gamma}\right] = \min_{a \in \mathcal{X}} Q^\pi_{\mathcal{M}}(x,a) + \frac{k}{1-\gamma} = J^\pi_{\mathcal{M}}(x) + \frac{k}{1-\gamma} = J^\pi_{\mathcal{M}'}(x),$$

and we can conclude that $J^\pi_{\mathcal{M}'} = J^*_{\mathcal{M}'}$ and π is also optimal in $\mathcal{M}'$.

(c) Repeating the same reasoning as in (b), we have that, in $\mathcal{M}''$,

$$\begin{aligned}
J^\pi_{\mathcal{M}''}(x) &= \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \mathrm{c}''_t \mid \mathrm{x}_0 = x\right] \\
&= \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t k \mathrm{c}_t \mid \mathrm{x}_0 = x\right] \\
&= k\mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t \mathrm{c}_t \mid \mathrm{x}_0 = x\right] \\
&= kJ^\pi_{\mathcal{M}}(x),
\end{aligned}$$

again due to the linearity of the expectation. Note also that

$$\min_{a \in \mathcal{X}} Q^\pi_{\mathcal{M}''}(x,a) = \min_{a \in \mathcal{X}} kQ^\pi_{\mathcal{M}}(x,a).$$

For $k \geq 0$, we can conclude that $J^\pi_{\mathcal{M}''} = J^*_{\mathcal{M}''}$ and, in that case, π is also optimal in $\mathcal{M}''$. For $k < 0$, however, the conclusion does not hold in general.

Question 4. (5.5 pts.)

Consider once again the MDP $\mathcal{M}$ defined in the previous shaded box, and suppose that—following a uniformly random policy π—the agent observes the trajectory

$$\tau = \{A, b, 1.0, B, b, 1.0, B\}.$$

Further consider the estimate for Q^π

$$\hat{\boldsymbol{Q}} = \begin{bmatrix} 0.63 & 1.17 & 1.23 \\ 1.24 & 1.25 & 1.01 \\ 0.03 & 0.25 & 0.12 \end{bmatrix}.$$

(a) **(1.5 pts.)** Perform one update of SARSA to $\hat{Q}(A, b)$. Use a step-size of $\alpha = 0.1$.

(b) **(1.5 pts.)** Perform one update of Q-learning to $\hat{Q}(A, b)$ (consider the original values of $\hat{Q}$). Again, use a step-size of $\alpha = 0.1$.

(c) **(1.5 pts.)** *Importance sampling* is a technique that allows samples from a distribution p to be used to compute the expectation with respect to a distribution q, taking advantage of the fact that

$$\mathbb{E}_{\mathrm{u}\sim q}\left[f(\mathrm{u})\right] = \sum_u q(u)f(u) = \sum_u p(u)\frac{q(u)}{p(u)}f(u) = \mathbb{E}_{\mathrm{u}\sim p}\left[\frac{q(\mathrm{u})}{p(\mathrm{u})}f(\mathrm{u})\right].$$

We can use *importance sampling* to derive a version of SARSA to compute $Q^{\pi'}$ given transitions obtained using a policy π. Specifically, we can use the modified update

$$Q(x_t, a_t) \leftarrow Q(x_t, a_t) + \alpha\left(c_t + \gamma\frac{\pi'(a_{t+1} \mid x_{t+1})}{\pi(a_{t+1} \mid x_{t+1})}Q(x_{t+1}, a_{t+1}) - Q(x_t, a_t)\right).$$

Use the above rule to perform one update of $\hat{Q}(A, b)$ (consider the original values of $\hat{Q}$). Use the greedy policy with respect to $\hat{Q}$ as π'.

(d) **(1.0 pts.)** Is the version of SARSA proposed in (c) off-policy or on-policy? Why?

Solution 4.

(a) Performing a SARSA update to $\hat{Q}(A, b)$, we get

$$\hat{Q}(A, b) \leftarrow \hat{Q}(A, b) + \alpha(1.0 + 0.9\,\hat{Q}(B, b) - \hat{Q}(A, b)) = 1.27.$$

(b) Performing a Q-learning update to $\hat{Q}(A, b)$, we get

$$\hat{Q}(A, b) \leftarrow \hat{Q}(A, b) + \alpha(1.0 + 0.9 \min_{a'} \hat{Q}(B, a') - \hat{Q}(A, b)) = 1.24.$$

(c) Performing an update using SARSA with importance sampling yields

$$\hat{Q}(A, b) \leftarrow \hat{Q}(A, b) + \alpha\left(1.0 + 0.9\,\frac{0.0}{0.33}\hat{Q}(B, b) - \hat{Q}(A, b)\right) = 1.15,$$

where the 0.33 comes from the fact that π is taken as the uniform policy.

(d) The proposed algorithm is off-policy since it will compute the value of a policy π' that is different from the policy used to obtain the samples (policy π). In fact, we can see that the algorithm will converge to a function such that

$$\begin{aligned} Q(x, a) &= c(x, a) + \gamma \sum_{x' \in \mathcal{X}} \mathbf{P}(x' \mid x, a) \sum_{a \in \mathcal{A}} \pi(a' \mid x')\frac{\pi'(a' \mid x')}{\pi(a' \mid x')}Q(x', a') \\ &= c(x, a) + \gamma \sum_{x' \in \mathcal{X}} \mathbf{P}(x' \mid x, a) \sum_{a \in \mathcal{A}} \pi'(a' \mid x')Q(x', a'), \end{aligned}$$

which is the recursion for $Q^{\pi'}$.