Exercise 05 - Reinforcement Learning

German Shiklov - 317634517

Question 1:

(Part 1.1)

(Q.1.1) Input: η - Learning Rate, T - total number of episodes

P_L - probability win left, P_R - win right machine

Initialization: $P_0 \leftarrow \frac{1}{2}$

Procedure:

For each episode, $t = \{1, 2, .., T\}$

1. select stochastically (normal, uniform), where $\begin{cases} 1 - P_{t-1} & ; a_t = a_L \\ P_{t-1} & ; a_t = a_R \end{cases}$
2. r_t, reward update $\in \{$action - win: 1, action - lost: 0$\}$
3. compute gradient of log-policy, $\nabla_P \log \pi(a_t | P_{t-1}) = \begin{cases} \frac{1}{P_{t-1}} & \text{if } a_t = a_R \\ -\frac{1}{1-P_{t-1}} & \text{if } a_t = a_L \end{cases}$
4. Estimate policy gradient, $G_t = r_t \cdot \nabla_P \log \pi(a_t | P_{t-1})$
5. Update policy parameter, $P_t = P_{t-1} + \eta \cdot G_t$
6. Project P_T to ensure $P_T \in (0,1) \Rightarrow P_t = \max(\min(P_t, 1-\epsilon), \epsilon)$

Output: P_T, probability of choosing the "most beneficial" machine.

(Part 1.2)

(Q.1.2) when there are no states, and immediate rewards obtained, the eligibility trace acts binary and resets every timestamp. as the decision is independent of previous action.

then let $e_t(a) = \begin{cases} 1 & \text{if } a = a_t \\ 0 & \text{otherwise} \end{cases}$

then the update policy parameter,

$$P_t = P_{t-1} + \underset{LR}{\eta}\,(r_t - \underset{\text{bias, zero.}}{b_t})\, e_t(a_t)\, \nabla_P \log \pi(a_t | P_{t-1})$$

(Part 1.3)

(Q.1.3)

let $R(p) = P_L(1-p) + P_R \cdot p$, be the expected reward, for policy p.

let $\frac{dR}{dp} = \frac{d}{dp}\left[P_L(1-p) + P_R\, p\right] = -P_L + P_R$

where in the general case $(b \neq 0)$, $p_t = p_{t-1} + \eta(r_t - b)\frac{\partial \log \pi(a_t \mid p)}{\partial p}$,

for $b=0$, $p_t = p_{t-1} + \eta \cdot r_t \cdot \frac{dR}{dp} = p_{t-1} + \eta \cdot r_t\,(P_R - P_L)$

If $P_R > P_L$, the policy intends to increase p when r_t is 1.

Conversely, $P_R < P_L$, decreases when $r_t = 0$, promoting the left machine.

(Part 1.4)

The blue line is unable to converge and mostly has deviation around 0.4 to 0.7 as the LR is not able to overstep the learning curve and figure the difference between the machines.

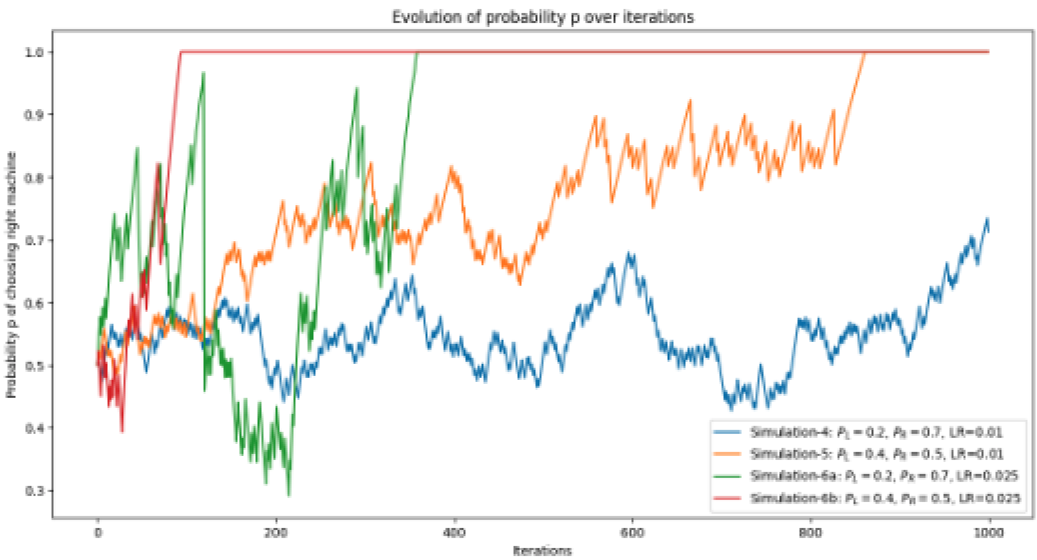
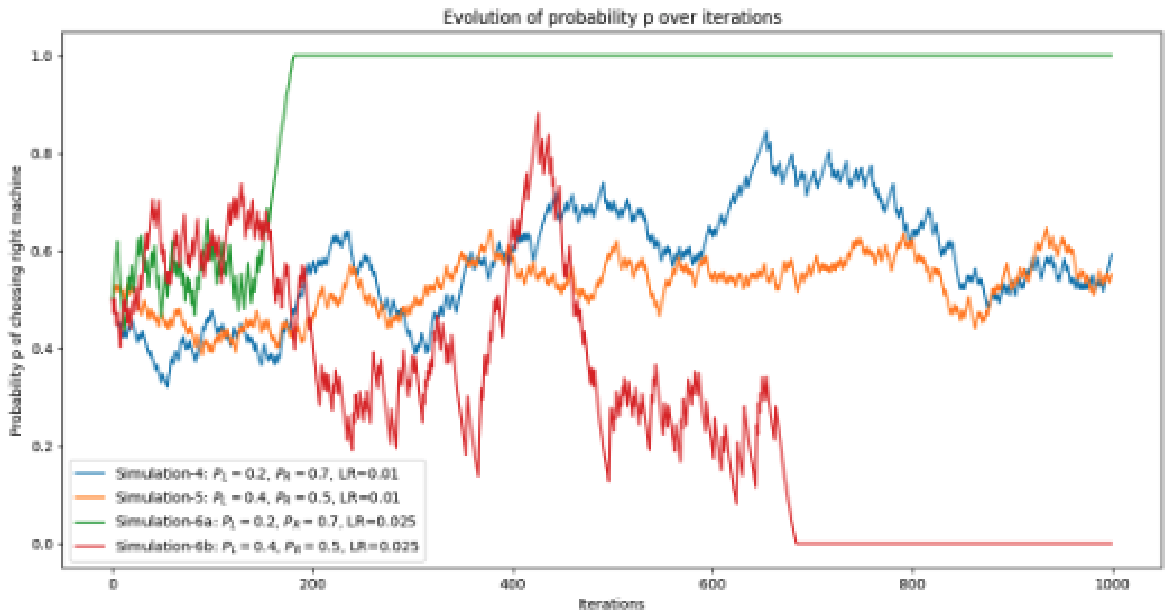


Figure 1. Simulation of P_L=0.2, P_R=0.7, LR = 0.01. The blue line doesn't seem to favor any machine after 1000 iterations.

(Part 1.5)
It's clearly visible that for both cases of LR 0.01, it is "not enough" to provide convergence and preferability to the machines, even though there's major differences between the PL, PR pairs. Figure 2 shows a scenario where the particularly "close" probabilities were difficult to converge, yet Figure 3 presents a scenario where the agent was able to converge and seemingly with a much shorter amount of steps.



"

Figure 2. Simulation of P_L=0.4, P_R=0.5, LR = 0.01. The orange line doesn't seem to favor any machine either, while comparing to the blue line like in the previous case..

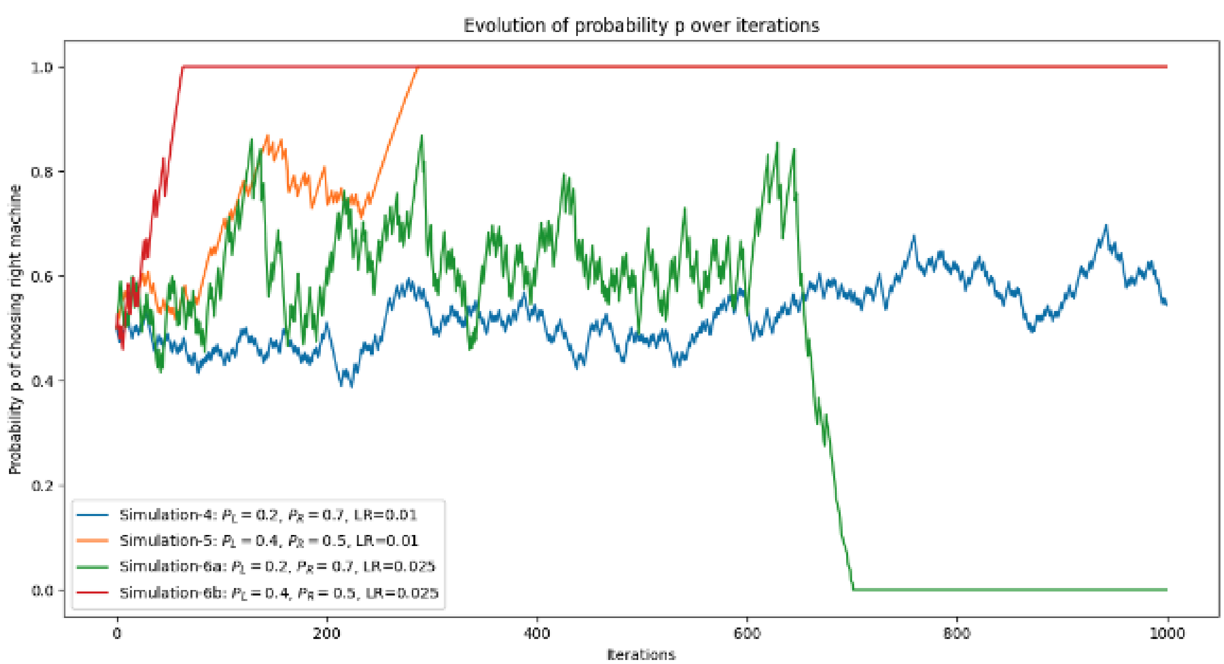


Figure 3. Simulation of P_L=0.4, P_R=0.5, LR = 0.01. The orange line has managed to favor a machine.

(Part 1.6)

The two cases of different probabilities, and higher learning rate (0.025), explicitly shows that it is able to converge after a while, compared to LR of 0.01. Though it would be expected that the green line would converge faster than the red, as there is a greater difference in probability between the two machines. Generally as a comparison to the lower LR, definitely the search has been made easier for simulation 6a-b, the red and green lines.

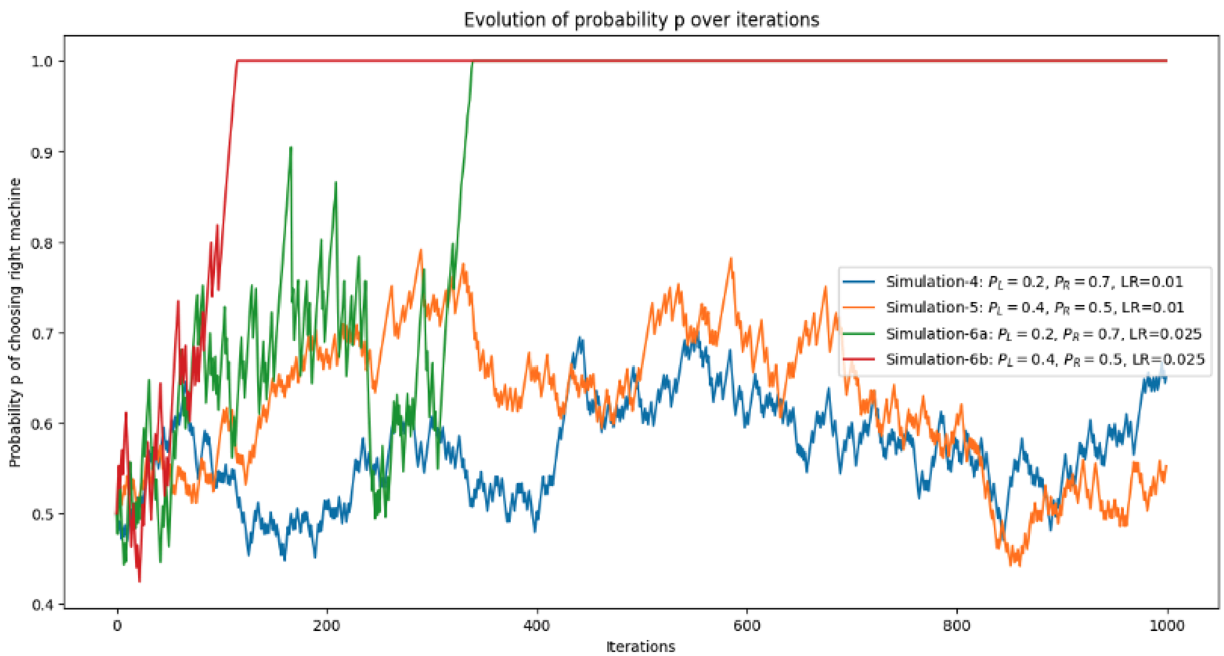


Figure 4. Simulation of P_L=0.2, P_R=0.7 as green line, and P_L=0.4, P_R=0.5, as red line, LR = 0.025. Both cases have managed to converge quite fast as the LR was great enough for yet greater difference between the left and right probabilities.

Question 2.1

(2.1.1) The optimal policy of $\pi_t(S,a) = \frac{1-\varepsilon}{4} + \underbrace{\varepsilon \cdot \delta_{a,a_t^*(S)}}$ $\lim_{\varepsilon \to 1} \frac{0}{4} + 1 \cdot \delta_{a,a_t^*(S)}$

$\lim_{\varepsilon \to 0} \frac{1-\varepsilon}{4} = \frac{1}{4}$ $\left\{ \begin{array}{l} \text{uniform dist pick for exploration} \\ \text{Any action chosen, not neccerially best} \\ \text{action to take} \end{array} \right.$

Is the exploitation part, biases the selection towards the current best known action.

- The policy is deterministic as we adjust the value of $\varepsilon \to 1$.

$$\pi^*(S) = \arg\max_{a \in A(S)} q_t^*(S,a)$$

1. where for the start state, S, $q^*(S, UP)$ is the must "first step" optimal. shortest path.
2. then for the other states, $q^*(S,a)$, reflects rewards of "shortest paths" to G, without cliffs.
3. For any states on the cliff, $q^*(S$, any action leading off the cliff$)$ - reflects very high penalty "for falling".

Let the optimal value function be $q(S_t, A_t) \leftarrow q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma \cdot \max_a q(S_{t+1}, a) - q(S_t, A_t) \right]$

(2.1.2)

As ε approaches 1, $\pi_t(S,a)$ converges to a purely greedy policy, $\lim_{\varepsilon \to 1} \pi_t(S,a) = \begin{cases} 1, & \text{if } a = \arg\max_{a \in A(S)} q_t^*(S,a) \\ 0, & \text{otherwise} \end{cases}$

Otherwise, $\lim_{\varepsilon \to 0} \pi_t(S,a) = \frac{1}{|A(S)|}$, makes it uniformly random.

A common practice is to start $\varepsilon = 0$, and raise to 1, to sharpen towards the optimal choice towards G.

Question 2.2.1

The optimal policy as a function of ϵ. Convergence occurs once the mean episodes go towards zero, obviously to some plateau near zero. This is due to the agent being able to learn the environment and the adjustment of the parameter, leading the decision towards the shortest path, minimizing the episode's exploration, thus transferring to the exploitation phase.

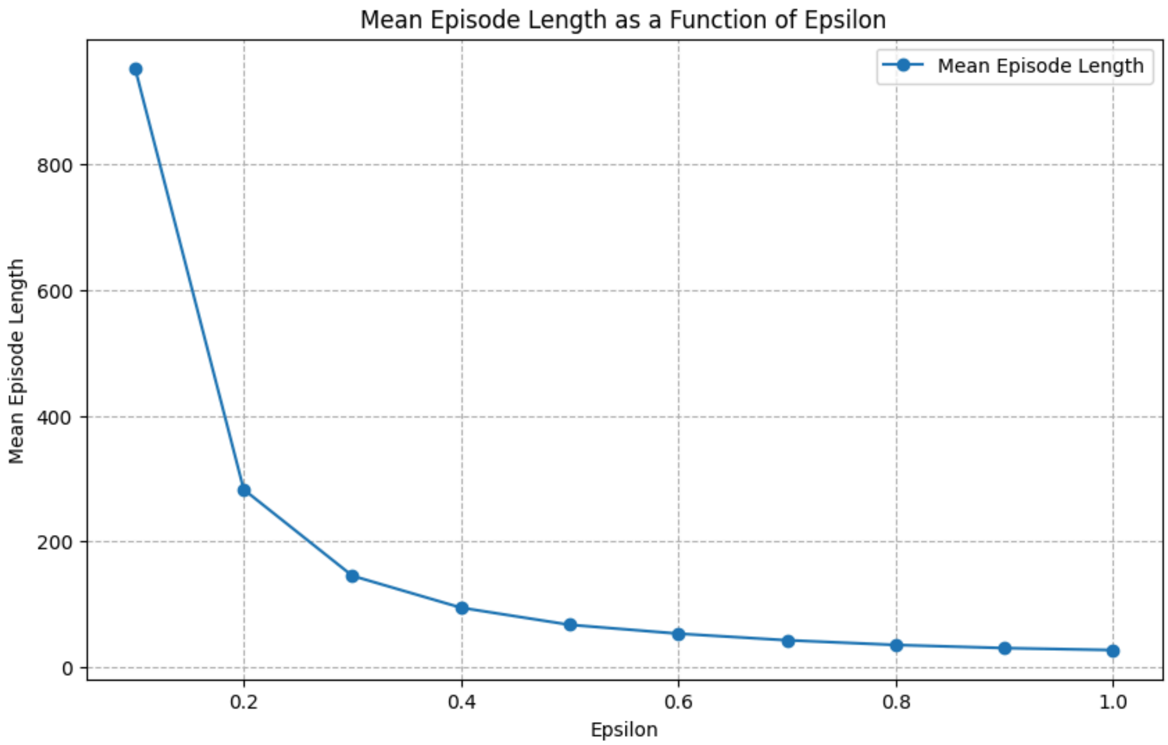


Figure 5. Optimal policy as a function of $\epsilon \rightarrow 1$. Converging over time.

Question 2.2.2

This plot demonstrates the mean total reward as a function of the episodes, thus making it apparent that for a very small ϵ, the decisions are more likely to be uniformly random, thus the blue lines finds it very difficult to converge with an order of magnitude, compared to greater epsilons. Interestingly there's not much of a difference between the 0.3 and 1 boundary, as it seems the grid(environment) is small enough to have the rewards to converge swiftly. Figure 6 shows how every epsilon which is below one has a plateau it reaches and cannot pass as the uniform-distribution is part of the computation. Sometimes it's worth it to keep the parameter small to allow exploration for greater grids.

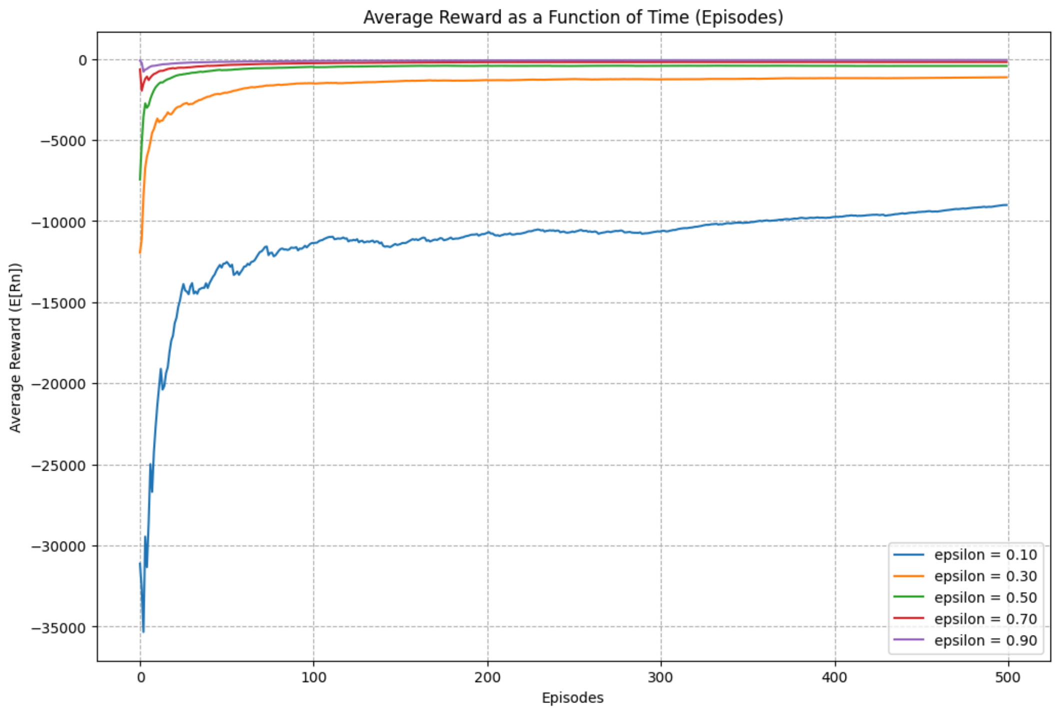
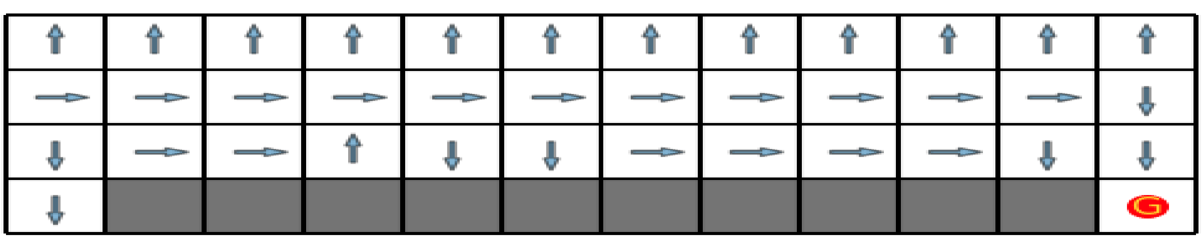


Figure 6. Mean total reward as a function of episodes, comparing different convergence rates for ϵ in [0.1-0.9]

Question 2 - Bonus

The motivation of using the greedy-policy is during learning to balance exploration and exploitation of the available states. Mainly, the benefit is particularly to emphasize the early learning process of the agent's knowledge of the environment is limited. As $\epsilon \to 0$, the phase of exploration is available, due to the "exploitation" decision being near zero chance, and thus making a uniform distribution between the four available states, up, down, right, left. Conversely the agent "less explores" when $\epsilon \to 1$ then the "random" choice disappears and the policy is more strict for choosing the likely best action. As the rewards manage to minimize, it will be beneficial to start changing to ϵ from 0 towards 1.



-12.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-11.0	-10.0	-9.0	-8.0	-7.0	-6.0	-5.0	-4.0	-3.0	-2.0	-1.0	0.0
-10.7	-10.2	-9.51	-8.7	-7.84	-6.93	-5.96	-4.98	-3.99	-3.0	-2.0	-1.0
-10.3											

Figure 7. Final results of optimal policy for 1000 iterations of episodes.