AIR5066/MDS5121: Reinforcement Learning **CUHKSZ**

Assignment 1

TA: Qibing Bai

Due Date: October 17th, 11:59 PM
Total points available: 100 pts.

Note: *Please note that external references are totally allowed only if you give an appropriate reference. There is no required format of reference. Please elaborate on your answers as well. (not just give a number etc.)*

Problem 1: Markov Decision Process I [30 pts.]

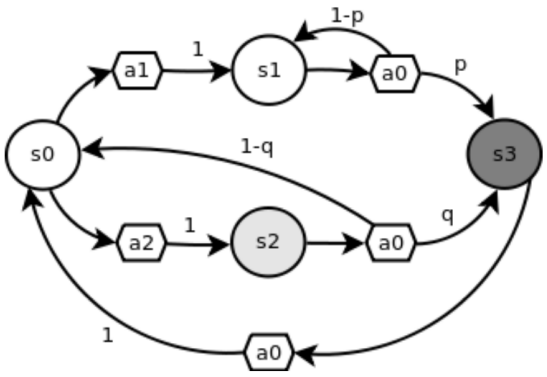

Figure 1: MDP for Problem 1. States are represented by circles and actions by hexagons. p, q denotes the transition probability and $p, q \in [0, 1]$. The reward is 10 for state s_3, 1 for state s_2 and 0 otherwise.

For this question, consider the infinite-horizon (where time $t = \infty$) MDP represented by Figure 1 with discount factor $\gamma \in [0, 1)$.

1. List all the possible policies [6 pts.].
2. Show the equation representing the optimal value function for each state, i.e. $V^*(s_0), V^*(s_1), V^*(s_2)$ and $V^*(s_3)$. [8 pts.]
3. Is there a value for p such that for all $\gamma \in [0, 1)$ and $q \in [0, 1], \pi^*(s_0) = a_2$? Explain. [8 pts.]
4. Is there a value for q such that for all $\gamma \in [0, 1)$ and $p > 0, \pi^*(s_0) = a_1$? Explain. [8 pts.]

Problem 2: Markov Decision Process II [30 pts.]

In this problem, we are going to see an alternative interpretation of the discount factor γ through bridging MDP with the stochastic shortest path problem (SSP). To simplify the math, we are going to redefine the MDP with the cost functions. That is, an MDP is the tuple $\mathcal{M} = \langle S, s_0, A, C, P \rangle$ where:

- S is the finite state space;
- $s_0 \in S$ is the initial state;
- A is the finite action space;
- $P(s' \mid s, a)$ represents the probability of $s' \in S$ be the resulting state of applying action $a \in A$ in state $s \in S$;
- $C(s, a) \in (0, \infty)$ is the cost of applying action $a \in A$ in state s.

With this, the optimal value function for infinite-horizon MDPs with discount factor $\gamma \in [0, 1)$ is

$$V^*(s) = \min_{a \in A} \left\{ C(s, a) + \gamma \sum_{s' \in S} P(s' \mid s, a) V^*(s') \right\}.$$

Notice that the min operator is used instead of max because the parametrization uses cost instead of reward.

Stochastic Shortest Path Problems (SSPs) are represented as the tuple $\mathcal{S} = \langle S, s_0, G, A, C, P \rangle$, where S, s_0, A, C and P are defined the same as in the MDP case and $G \subset S$ is the set of goal states. Both MDPs and SSPs have solutions as a policy, however, the solution policy in SSPs should instead minimize the expected cost to reach a goal state. We assume that for all state $s \in S$, there exists a path from s to $s_G \in G$ for the SSPs.

1. Explain why infinite-horizon MDPs need the discount factor γ. [4 pts.]
2. Explain why SSPs do not need the discount factor γ. [4 pts.]
3. Derive the optimal value function for SSPs, i.e. the equivalent of V^* for SSPs. [8 pts.]
4. Given an infinite-horizon MDP $\mathcal{M} = \langle S, s_0, A, C, P \rangle$ with discount factor $\gamma \in [0, 1)$ show how to translate $\mathcal{M}$ into a SSP $\mathcal{S} = \langle S', s_0, G, A, C', P' \rangle$. [14 pts.]

Problem 3: Bandits Problem [15 pts.]

Consider a bandit instance with 4 arms and running ϵ-greedy algorithm on this instance. The algorithms maintain an initial estimate of the reward mean of arm i, μ_i, and we denote the estimate at time t as $\hat{\mu}_i(t)$. The algorithm initialize $\hat{\mu}_i(1) = 0$, for all i. Then the algorithm observes the following sequence of actions and rewards, where A_t, R_t denote the action and reward at time t :

- $t = 1, A_1 = 1, R_1 = 3$.
- $t = 2, A_2 = 2, R_2 = 2$.
- $t = 3, A_3 = 1, R_3 = 1$.
- $t = 4, A_4 = 3, R_4 = 1$.
- $t = 5, A_5 = 2, R_5 = 2$.

1. Which of the actions was definitely exploratory? (Recall ϵ-greedy explore with probability ϵ). [8 pts.]
2. Which of the actions was possibly exploratory? [7 pts.]

Problem 4: Adaptive Explore-Then-Commit [25 pts.]

We consider a two-armed bandit problem where both arms are 1-subgaussian with means μ_1, μ_2, and the gap is $\Delta = |\mu_1 - \mu_2|$. For the following question, we will analyze a more advanced, **adaptive** version of the ETC algorithm. Instead of exploring for a fixed number of steps, the algorithm will stop exploring based on the data it has collected.

Let $\hat{\mu}_i(k)$ be the empirical mean of arm i after k pulls. The adaptive algorithm works as follows:

1. **Explore:** For $k = 1, 2, 3, \ldots, \lfloor T/2 \rfloor$, pull each arm once (alternately).
2. **Check Condition:** After each arm has been pulled k times, calculate the confidence interval radius:
$$w(k) = \sqrt{\frac{2\log(T)}{k}}$$
3. **Stop & Commit:** If $|\hat{\mu}_1(k) - \hat{\mu}_2(k)| > 2w(k)$, stop exploring. Let K be the number of pulls of *each arm* at which this condition was met. Commit to the arm with the higher empirical mean $\hat{\mu}_i(K)$ for the remaining $T - 2K$ steps. If the condition is never met, explore for all $\lfloor T/2 \rfloor$ steps for each arm.

Your goal is to show that this adaptive strategy leads to a regret bound that improves when Δ is large.

1. **Bounding the Exploration Duration**. First, show that if the algorithm stops, the exploration phase is unlikely to be very long for a large gap Δ. Prove that the stopping time K is bounded by:
$$K \leq \frac{32 \log T}{\Delta^2} + 1$$
with high probability. [15 pts.]

   *Hint: Assume the algorithm has **not** stopped before step k. This means $|\hat{\mu}_1(k-1) - \hat{\mu}_2(k-1)| \leq 2w(k-1)$. Use the reverse triangle inequality and the definition of the confidence radius to relate this to Δ, and rearrange to get a bound on k.*

2. **Deriving the Final Regret Bound**. The total regret (ignoring small probability error events) is dominated by the exploration phase, where the regret is given by $K \cdot \Delta$. Using your result from part (1), show that the total regret is bounded by:
$$\bar{R}_T \leq \Delta + \frac{C \log T}{\Delta}$$
for some universal constant C. [10 pts.]