Birla Institute of Technology & Science, Pilani
Work Integrated Learning Programmes Division
First Semester 2024-2025

Mid-Semester Test
(EC-2 Regular)

Course No.	: AIMLCZG512
Course Title	: Deep Reinforcement Learning
Nature of Exam	: Closed Book
Weightage	: 30%
Duration	: 2 Hours
Date of Exam	: **** (FN/AN)

No. of Pages = 3
No. of Questions = 6

Note to Students:
1. Please follow all the *instructions for candidates* given on the cover page of the answer book.
2. All parts of a question should be answered consecutively. Each answer should start from a fresh page.
3. Assumptions made if any, should be stated clearly at the beginning of your answer.

Q1) [4+1 = 5 Marks]

(a) Consider the problem of designing a reinforcement learning agent to be employed by India Post to deliver the parcels to the consignee's doorsteps. A robot picks k consignments from the post office, delivers them to the consignees and reaches back to the post office. Given the choice of reward functions (i) and (ii), which will you choose [in one word, 1.0 M] and why [in not more than three sentences, 1.5 M, choose your statements to make a clear argument]? Why would you not choose the other option you have not selected? [1.5 M] Explain. The agent is expected to safely deliver all the assigned consignments while maximising the number of deliveries on a given day. Not all the consignees are living close to the post office.

Reward (i) Agent receives -1 for each failure in delivering a consignment to the correct consignee and 0 for a successful delivery.

Reward (ii) Agents receive +1 for each successful delivery to the consignee and 0 for each failure.

(b) Consider the problem in (a) to be solved using the Montecarlo method. What will be considered an episode in the given situation, each k deliveries or all the deliveries in a day? Give reasons for your answer [1.0 M].

Q2) [3+2= 5 Marks]

(a) Consider a k -armed bandit problem with k = 3 actions, denoted 1, 2 and 3. Consider applying to this problem a bandit algorithm using "ε-greedy action selection, sample-average action-value estimates, and initial estimates of Q1 (a) = 0, for all a.
Suppose the initial sequence of actions and rewards is
(A1=2,R1=4), (A2=1, R2=4), (A3=3, R3=2), (A4=2, R4=2), (A5=3, R5=0,) (A6=1, R6=4).
On some of these time steps, the ε- case may have occurred, causing an action to be selected at random. At which time steps did this definitely occur? At what time could this possibly have happened? Provide your response in a table as below: [3 Marks].

Timesteps	Your Answer	Explanation
1		
2		

3		
4		
5		
6		

(b) Consider at time steps 7 and 8 (i.e. continuing from the previous example) we use UCB to select the next actions. What those actions would be? Assume the reward at time steps 7 and 8 are 4 each, irrespective of the chosen actions. Use the value of c as 2 in your calculations [2 calculations x 1 = 2 Marks].

Q3) [2+3+1 = 6 Marks]

Consider an automated truck climbing on a slope, which recharges itself using solar panels. Three states represent its charge level – low, medium and high. If it accelerates all through the way, it moves across all three states (low to medium to high) and stays high with a probability of 0.6, and it stays where it is with a probability of 0.4. If it does not accelerate, it slides down the slope to low with a probability of 0.5 and stays where it is with a probability of 0.5. Accelerating uses two units of energy per time step, while staying on high or medium gains 3 units of energy via panel. Staying in low does not gain energy. The automated truck has to gain as much energy as possible. State and make assumptions only for cases not covered in this description.

For the above scenario,

a) Draw the MDP graphically. Explicitly mention the transition probabilities and rewards.[2 Marks]
b) Using synchronous dynamic programming and value iteration, determine the optimal values of states. Use 1 as the initial value of states and 0.3 as the discount factor. Solve only for two iterations. [3 Marks]
c) Using the results of b), find if there are any changes in the optimal policy. [1 Marks]

Q4) [4+ 2 = 6 Marks]

In off-policy learning, learning the agent involves using a behaviour policy b and a target policy π. The agent can be in states S1 or S2 at any point in time. The actions Up and Down are available from all the states. The behavior and target policies are as below:

Behaviour Policy b	Up	Down
S1	0.4	0.6
S2	0.6	0.4

Target Policy π	Up	Down
S1	0.8	0.2
S2	0.2	0.8

Consider the following episode generated by the behaviour policy:

S1 - Up - (+4) - S2 - Down - (+6) - S1 - Down - (+8) - S2 - Down - (+16) - S1 - Down - (+2) - S2 - Down - (+4) - S1 - Up - (+4) - S2 - Down - (+16)

(a) Estimate the value of the state-value (S1, Down) and (S1, Up) using **every visit** estimate. Show all the steps. Assume the discount factor to be 1. [2 + 2 =4 Marks]

(b) Assume the target policy π for S1 is updated after the estimation in (a) using ε-greedy. What will be π(Up | S1) and π(Down|S1) assuming ε = 0.4. [2 Marks]

Q5) [3 + 2 = 5 Marks]

(a) Provide an example of an RL application for which the solution based on Dynamic Programming is more appropriate than the Monte Carlo Method. [Application + Explanation in 2 statements - 1.5 Marks].Also, provide an application for which the solution based on Montecarlo is more appropriate than Dynamic Programming. [Application + Explanation in 2 statements -1.5 Marks].

(b) Provide a comparison of Policy Iteration and Value Iteration Methods. [Give a 2 point comparison with most appropriate points - 2 Marks]

Q6) [3 Marks]

Recollect the recycling robot example (from the Textbook) discussed in the class, whose dynamics is reproduced below. Write down the bellman expressions for v_π(high), q_π(high, search), v_*(high), q_*(high, search) that suit the following dynamics, i.e. make the bellman equations to be very specific to this case.

s	a	s'	$p(s' \mid s, a)$	$r(s, a, s')$
high	search	high	α	r_{search}
high	search	low	$1 - \alpha$	r_{search}
low	search	high	$1 - \beta$	-3
low	search	low	β	r_{search}
high	wait	high	1	r_{wait}
high	wait	low	0	-
low	wait	high	0	-
low	wait	low	1	r_{wait}
low	recharge	high	1	0
low	recharge	low	0	-

SOLUTION

① (a) Reward 1 : −1 for failure
0 for success

Reward 2 : +1 for success
0 for failure.

Ans : Reward 2

Why ? It positively reinforces correct deliveries. Encourages agent to maximize correct actions & aligns with goal of maximi-zing daily deliveries.

Why not reward 1 ? It does not give credit for successful deliveries, making it hard for agent to distinguish b/n neutral & good performance, thereby slowing down learning.

(b) Assuming Monte carlo method.

Answer : All deliveries in a day.

Reason : Episode in MC method should capture the complete experience leading to cumulative reward. Since the objective is to optimize deliveries per day, the entire day's deliveries define one full episode.

② (a) K-armed Bandit pbm with k = 3 [1, 2, 3].

(A1 = 2, R1 = 4), (A2 = 1, R2 = 4), (A3 = 3, R3 = 2), (A4 = 2, R4 = 2),

(A5 = 3, R5 = 0), (A6 = 1, R6 = 4).

Timestep	Answer	Timestep	Answer
1	possibly	4	possibly
2	Definitely	5	Definitely
3	Definitely	6	Greedy.

(b) UCB

$$UCB(a) = Q(a) + C\sqrt{\frac{\ln t}{N(a)}}$$

At T=7

$Q(1) = (4+4)/2 = 4,\ N=2$

$Q(2) = (4+2)/2 = 3,\ N=2$

$Q(3) = (2+0)/2 = 1,\ N=2$

$UCB\ 1 = 4 + 2\sqrt{\ln(7)/2} = 4 + 1.56 = 5.56$

$UCB\ 2 = 3 + 2\sqrt{\ln(7)/2} = 3 + 1.56 = 4.56$

$UCB\ 3 = 1 + 2\sqrt{\ln(7)/2} = 1 + 1.56 = 2.56$

T7 Action = 1

At T=8

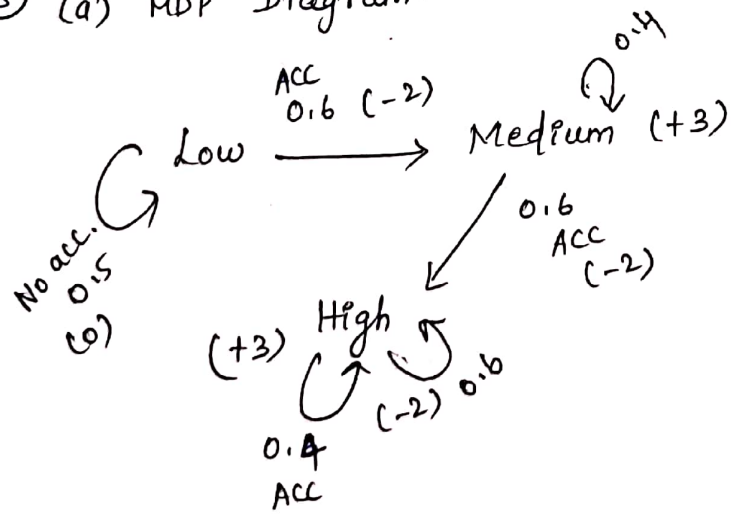
$Q(1) = (4+4+4)/3 = 4$; $N(1) = 3$ Now.

$UCB(1) = 4 + 2\sqrt{(\ln 8)/3} = 4 + 1.23 = 5.23$

UCB for 2 & 3 are same.

T8 action = 1

③ (a) MDP Diagram.



(b) Dynamic programming & value iteration.

Initial value = 1.

$\gamma = 0.3$ [Find for 2 iterations].

$V=1$ for all states

Iteration 1 : $V(s) \leftarrow \max_a \sum_{s'} P(s'|s,a)[R(s,a,s') + \gamma \cdot V(s')]$

Low.

↳ Accelerate [Low → Medium].

$1[(-2) + 0.3\,(V_{medium})]$

$= -2 + 0.3(1) = -1.7$

↳ No acceleration

$= 0.5\,(0 + 0.3(V_{low})) + 0.5\,(0 + 0.3(V_{low}))$

$= 0.5\,(0.3) + 0.5\,(0.3) = 3.$

$\boxed{\text{Max}(-1.7, 3) = 3.}$ $\boxed{V_{low} = 3}$

Medium

↳ Accelerate [Medium → High / Medium]

$1\,(-2 + 0.3\,(0.6 \times 1 + 0.4 \times 1)) = -1.7$

(0.6 ↓ high; 0.4 ↓ Medium)

↳ No acceleration

$(+3) + 0.3\,(0.5 \times 1 + 0.5 \times 1) = 3.3$

$\text{Max}(-1.7, 3) = 3$

$\boxed{V_{medium} = 3}$

High

↳ Accelerate : Stay → $0.6\,(-2 + 0.3 \times 1) + 0.4\,(3 + 0.3 \times 1)$

$= 0.6\,(-1.7) + 0.4\,(3.3)$

$= -1.02 + 1.32 = 0.3$

$\boxed{V_{high} = 0.3}$

Repeat with these values in Iteration 2.

(c) Policy change

Yes, optimal policy changes from acceleration to no acceleration in medium / high due to higher reward from staying.

(4) s_1, up (+4) - s_2, down (+6) - s_1, down (+8) - s_2, down (+16) - s_1, down (+2) - s_2, down (+4) - s_1, up (+4) - s_2, down (+16)

(a) s_1, down $[\gamma = 1]$.

After 1st : $+8 + 1(16) + 1^2(2) + 1^3(4) + 1^4(4) + 1^5(16)$

$= 8 + 16 + 2 + 4 + 4 + 16 = \boxed{50}$

After 2nd : $2 + 1(4) + 1^2(4) + 1^3(16) = 2 + 4 + 4 + 16 = \boxed{26}$

Avg $= \frac{50 + 26}{2} = \frac{76}{2} = \boxed{38}$

(s_1, up)

After 1st : $4 + 6 + 8 + 16 + 2 + 4 + 4 + 16 = 60$

After 2nd : $4 + 16 = 20$

Avg $= \frac{60 + 20}{2} = \boxed{40}$

(b) ϵ-greedy

$\pi(s_1)$. $\epsilon = 0.4$

Find $\pi(up|s_1)$ & $\pi(down|s_1)$

$(s_1, up) = 40$, $(s_1, down) = 38$

$(s_1, up) \rightarrow$ greedy.

$$\pi(a|s) = \begin{cases} 1 - \epsilon + \frac{\epsilon}{n} & a = \text{greedy} \\ \epsilon/n & a \neq \text{greedy.} \end{cases}$$

$\pi(up|s_1) = 1 - 0.4 + \frac{0.4}{2} = 0.8$

$\pi(down|s_1) = \frac{0.4}{2} = 0.2$

5 (a)

(i) DP better

Elevator scheduling - full model is available, transition prob. are known.

(ii) MC better

Game playing agent - no model available, learns from situation / play.

(b) Direct qn.

6 States : high, low

Actions : Search, wait, recharge

$$V_\pi(high) = (\pi(search|high) * q_\pi(high, search)) + (\pi(wait|high) * q_\pi(high, wait))$$

$$q_\pi(high, search) = 0.9\,(R + \gamma * V_\pi(high)) + 0.1\,(-3 + \gamma * V_\pi(low))$$

$$V(high)^* = \max_a q^*(high, a)$$

$$q(high, search)^* = [0.9 * (R + \gamma * V^*(high))] + [0.1 * (-3 + \gamma * V^*(low))]$$