

Reinforcement Learning

Assignment 2 (MDP and Dynamic Programming)

Prashant Pathak (MT19051)

[Question 1](#)

[Question 2](#)

[Question 3](#)

[3.15](#)

[3.16](#)

[Question 4](#)

[Question 5](#)

[Question 6](#)

[Question 7](#)

[Question 8](#)

[Question 9](#)

[Question 10](#)

[Question 11](#)

[Question 12](#)

[Question 13 \(a\)](#)

[Question 13 \(b\)](#)

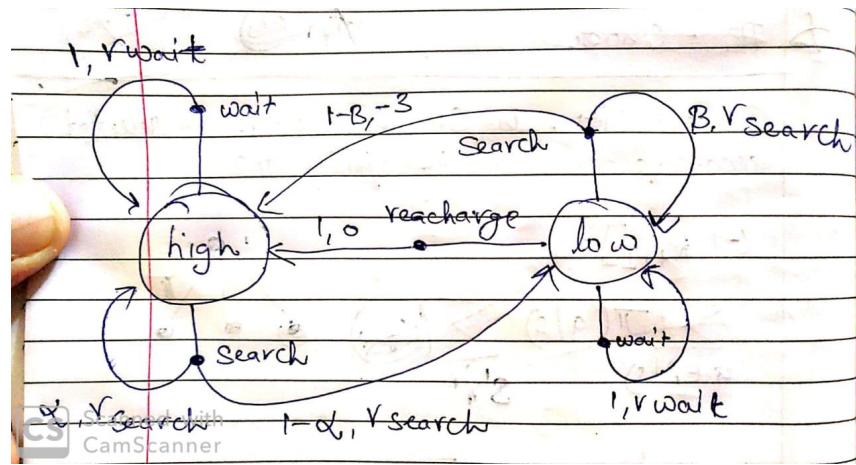
[Question 13 \(c\)](#)

[Question 14](#)

[Question 15](#)

Question 1

Below we have the MDP of the problem.



For making the table we have to consider all the possibility of the current states (s), actions that can be taken(a), next sates (s') and rewards(r).

As explained in the book, r_{search} and r_{wait} are the expected reward that an agent receives by taking a particular action. So to actually make the table off $p(s',r|s,a)$ we have to consider the distribution (i.e. PMF) of the reward (i.e. r_{search} and r_{wait}) and include all the individual values for making the table. For instance, if r_{search} can take values $r_{\text{search}} 1, r_{\text{search}} 2$ and $r_{\text{search}} 3$ with probability p_1, p_2 and p_3 then all these values will come in the table.

For making the below table I have considered r_{search} and r_{wait} as a constant reward.

Table for $p(s',r | s,a)$

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

low	wait	low	r_{wait}	1
low	recharge	high	0	1
low	recharge	low	---	0

Question 2

We have to solve this RL problem by solvinf the linear equations. So First we have found the linear equation and solved it using matrix opeation.

1	2	3	4	5
A			B	
6	3	8	9	10
7	0		B +5	
			y	
			B'	
			B' _{max}	
	14			25
	22			

Page No.	
Date	

→ States all grid cell are states we can denote them by,
 s_1, s_2, \dots, s_{25}

→ Action → Four action at each cell. N, S, E, W i.e.
 North, South, East, West

→ Reward → from ~~the~~
 $r(s, a)$

→ Action that take off grid reward is -1

→ Action from A to A' = 10 (despite any action taken)

→ Action from B to B' = 5 (despite any action taken)

We have to find the value function for each state given the policy that all actions are equiprobable

$$V_\pi(s) = \sum_a \pi(a|s) \sum_{s', r} p(s', r | s, a) [r + \gamma V_\pi(s')]$$

for all $s \in S$

Here from a state and action pair we can go to only one other state & get definite reward.

Equation for each state:

$$V(S_1) = \frac{1}{4} [-1 + 0.9 * V(S_1)]$$

Page No. _____
Date _____

$$V(S_2) = 0.25 [-1 + 0.9 * V(S_1)] + 0.25 [0 + 0.9 * V(S_2)] + 0.25 [0 + 0.9 * V(S_6)]$$

$$= -0.5 + 0.45 V(S_1) + 0.225 V(S_2) + 0.225 V(S_6)$$

$$0.55 V(S_1) - 0.225 V(S_2) - 0.225 V(S_6) + 0.5 = 0$$

$\hookrightarrow \Sigma_2 \quad (1)$

$$V(S_2) = 10 + 0.9 V(S_{22})$$

$$V(S_2) - V(S_{22}) - 10 = 0 \quad \rightarrow \Sigma_2 \quad (2)$$

$$V(S_3) = 0.25 [0 + 0.9 V(S_2)] + 0.25 [-1 + 0.9 V(S_3)] + 0.25 [0 + 0.9 V(S_4)] + 0.25 [0 + 0.9 V(S_6)] - \Rightarrow 0.225 V(S_2) + 0.225 V(S_3) + 0.225 V(S_4) + 0.225 V(S_6) - 0.25$$

$$0.775 V(S_3) - 0.225 V(S_2) - 0.225 V(S_4) - 0.225 V(S_6) + 0.25 = 0 \quad \rightarrow \Sigma_3 \quad (3)$$

$$V(S_4) = 5 + 0.9 V(S_{12}) \quad \rightarrow \Sigma_4 \quad (4)$$

$$V(S_5) = 0.55 V(S_5) - 0.225 V(S_4) - 0.225 V(S_{10}) + 0.5 = 0 \quad \rightarrow \Sigma_5 \quad (5)$$

$$V(S_6) = 0.25 [-1 + 0.9 V(S_6)] + 0.25 [0 + 0.9 V(S_1)] + 0.25 [0 + 0.9 V(S_{11})] + 0.25 [0 + 0.9 V(S_7)]$$

$$V(S_6) = -0.25 + 0.225 V(S_6) + 0.225 V(S_1) + 0.225 V(S_{11}) + 0.225 V(S_7)$$

$$0.775 V(S_6) - 0.225 V(S_1) - 0.225 V(S_{11}) - 0.225 V(S_7) + 0.25 = 0 \quad \rightarrow \Sigma_6 \quad (6)$$

$$V(S_7) = 0.225 V(S_2) + 0.225 V(S_8) \\ + 0.225 V(S_{12}) + 0.225 V(S_6) - \varepsilon_v(7)$$

Page No.
Date

$$V(S_8) = 0.225 V(S_9) + 0.225 V(g) + 0.225 V(S_{13}) \\ + 0.225 V(S_7) - \varepsilon_v(8)$$

$$V(S_9) = 0.225 V(S_4) + 0.225 V(10) + 0.225 V(S_{14}) \\ + 0.225 V(S_8) - \varepsilon_f(9)$$

$$V(S_{10}) = 0.225 V(S_5) + 0.225 V(10) - 0.25 + \\ 0.225 V(S_{15}) + 0.225 V(S_9) - \varepsilon_f(10)$$

$$V(S_{11}) = 0.225 V(S_6) + 0.225 V(S_{12}) + 0.225 V(S_{14}) \\ + 0.225 V(S_{11}) - 0.25 - \varepsilon_v(11)$$

$$V(S_{12}) = 0.225 V(S_7) + 0.225 V(S_{13}) + 0.225 V(S_7) \\ + 0.225 V(S_{11}) - \varepsilon_v(12)$$

$$V(S_{13}) = 0.225 V(S_8) + 0.225 V(S_{14}) + 0.225 V(S_{18}) \\ + 0.225 V(S_{12}) - \varepsilon_v(13)$$

$$V(S_{14}) = 0.225 V(S_9) + 0.225 V(S_{15}) + 0.225 V(S_{19}) \\ + 0.225 V(S_{13}) - \varepsilon_v(4)$$

$$V(S_{15}) = 0.225 V(S_{10}) + 0.225 V(S_{15}) - 0.25 + \\ 0.225 V(S_{20}) + 0.225 V(S_{14}) - \varepsilon_v(15)$$

$$V(S_{16}) = 0.225 V(S_{11}) + 0.225 V(S_{17}) + 0.225 V(S_{21}) \\ + 0.225 V(S_{16}) - 0.25$$

$$V(S_{17}) = 0.225 V(S_{12}) + 0.225 V(S_{18}) + 0.225 V(S_{22}) \\ + 0.225 V(S_{16})$$

$$V(S_{18}) = 0.225V(S_{13}) + 0.225V(S_{19}) \\ 0.225V(S_{23}) + 0.225V(S_{17}) \quad \text{Page No.} \quad \text{Date - } \epsilon_1(18)$$

$$V(S_{19}) = 0.225V(S_{14}) + 0.225V(S_{20}) \\ 0.225V(S_{24}) + 0.225V(S_{18}) \quad \epsilon_1(19)$$

$$V(S_{20}) = 0.225V(S_{15}) + 0.225V(S_{21}) - 0.25 \\ + 0.225V(S_{25}) + 0.225V(S_{19}) \quad \epsilon_1(20)$$

$$V(S_{21}) = 0.225V(S_{16}) + 0.225V(S_{22}) \\ + 0.225V(S_{21}) - 0.5 \quad \epsilon_1(21)$$

$$V(S_{22}) = 0.225V(S_{17}) + 0.225V(S_{23}) + \\ 0.225V(S_{22}) - 0.25 + 0.225V(S_{21}) \quad \epsilon_1(22)$$

$$V(S_{23}) = 0.225V(S_{18}) + 0.225V(S_{24}) + \\ 0.225V(S_{23}) - 0.25 + 0.225V(S_{22}) \quad \epsilon_1(23)$$

$$V(S_{24}) = 0.225V(S_{19}) + 0.225V(S_{25}) \\ + 0.225V(S_{24}) - 0.25 + 0.225V(S_{23}) \quad \epsilon_1(24)$$

$$V(S_{25}) = 0.225V(S_{20}) + 0.225V(S_{24}) \\ + 0.225V(S_{25}) - 0.5 \quad \epsilon_1(25)$$

Now we have 25 variable & 25 equation
 we can solve it to find value of
 value function.

$$\cancel{V = C} \quad V * C = \cancel{X}$$

6  Scanned with CamScanner \vec{V} = row vector of ~~state~~ state $R^{25 \times 1}$

$$C = R^{25 \times 25}$$

$$X = R^{25 \times 1}$$

Page No.

Date

$$\text{so } V = C^{-1} X$$



Scanned with
CamScanner

We use the equation above and coded it. Can be viewed in Question2.py

Values of the state after solving the equation:

```
array([[ 3.3,  8.8,  4.4,  5.3,  1.5],  
       [ 1.5,  3. ,  2.3,  1.9,  0.5],  
       [ 0.1,  0.7,  0.7,  0.4, -0.4],  
       [-1. , -0.4, -0.4, -0.6, -1.2],  
       [-1.9, -1.3, -1.2, -1.4, -2. ]])
```

Question 3

3.15

3.15 From Eq 3.8 of book we have

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots$$

$$= \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

Now suppose we add a constant reward γ_c to all the reward so now

$$G'_t = R_{t+1} + \gamma G_t$$

$$G'_t = (R_{t+1} + \gamma_c) + \gamma (R_{t+2} + \gamma_c) + \gamma^2 (R_{t+3} + \gamma_c)$$

$$= R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots$$

$$+ \gamma_c + \gamma \gamma_c + \gamma^2 \gamma_c$$

$$= \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} + \sum_{k=0}^{\infty} \gamma^k \gamma_c$$

$$= G_t + \frac{\gamma_c}{1-\gamma}$$

This is from 3.8

\rightarrow (Sum of G.P (infinite when $\gamma < 1$))

Now we have to calculate this value.

at state



Scanned with
CamScanner

$$V_{\pi}(s) = E[G_t | S_t = s]$$

Page No.

Date

$$= E\left[\left(G_t + \frac{V_c}{1-\gamma}\right) | S_t = s\right]$$

* Using linearity of expectation.

$$= E[G_t | S_t = s] + E\left[\frac{V_c}{1-\gamma} | S_t = s\right]$$

As $\frac{V_c}{1-\gamma}$ is a constant

$$= E[G_t | S_t = s] + \frac{V_c}{1-\gamma}$$

So we can say that if adding a constant ~~does~~ change the final state values by a constant $V_c = \frac{C}{1-\gamma}$

So relative values are not changed between states.

3.16

3.16 - In case of episodic tasks

$$G_t = R_{t+1} + R_{t+2} + R_{t+3} + \dots + R_T$$

where T is terminating time.

Now if we here add a constant C to all the reward

$$\begin{aligned} G_t &= R_{t+1} + C + R_{t+2} + C + \dots + R_{T-1} + C \\ &= R_{t+1} + R_{t+2} + \dots + R_T + C + \dots + C \end{aligned}$$



Scanned with
CamScanner

$$G_t' = G_t + \text{Length } \delta$$

Episode $\times C$

Page No.	1
Date	1

→ Here length Episode is Not fixed.

$$\begin{aligned} \mathbb{E} V_T(s) &= \mathbb{E}[G_t' | S_t=s] \\ &= \mathbb{E}[G_t + \text{length } \delta \text{ Episode. } \times C | S_t=s] \\ &= \mathbb{E}[G_t | S_t=s] + \underline{\text{length } \delta \text{ EpVC}} \end{aligned}$$

→ Here the value that is added to each episode is not constant so it can affect the relative ordering of the task.

→ Suppose by adding a Constant C to -ve reward it is ~~make~~ positive the our makes it positive the it would affect how agent moves so by both sign & relative values are important in episodic task.



Scanned with
CamScanner

Question 4

In this, we have to find optimal values function and optimal policy for Previous grid word in question 2. But we have to find it by solving non-linear equations.

Q4

for calculating the optimal ~~values~~ equation for each state we will need below formula.

$$V_{\pi}(s) = \max_a \sum_{s' r} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

So for each state we have to find the immediate reward & the ~~next~~ boot strap value of next state.



Scanned with
CamScanner

~~Ex for State 2~~

$$V_x(s_1) = \max_a \left[(-1 + 0.9 V_x(s_1)) \underbrace{,}_{a}, (-1 + 0.9 V_x(s_2)) \underbrace{,}_{b} \right] \\ \rightarrow (0.9 * V_x(s_2), 0.9 * V_x(s_1))$$

~~Similarly for~~

~~Q.~~ \rightarrow

a \rightarrow It is the case when I move west. receive reward of -1 & reach $V_x(s_1)$ s_1

b \rightarrow It is when we go North, receive reward of -1 & reach state s_1

c \rightarrow It is when we move east & receive 0 & reach s_2

d \rightarrow It is when Agent move South & receive 0 & reach s_6

likewise we have equation for all the state ..

Equation can be found in code Notebook then we solved this equation using `scipy.optimize.fsolve` to solve equation.

To find the optimal policy we can use below formula

$$a^* = \operatorname{arg\max}_a \sum_{s' \in R} p(s', V(s)) [V + V_x(s')]$$

See Question4.py code for detail.

Optimal Values of the state after solving the non linear equation (It matches that of fig 3.5 of book):

```
[[22. 24.4 22. 19.4 17.5]
 [19.8 22. 19.8 17.8 16. ]
 [17.8 19.8 17.8 16. 14.4]
 [16. 17.8 16. 14.4 13. ]
 [14.4 16. 14.4 13. 11.7]]
```

Optimal Policy after solving the equation:

Probability of each action in Cell(i, j) are: [N E S W]

Probability of each action in Cell(0 , 0) are: [0. 1. 0. 0.]

Probability of each action in Cell(0 , 1) are: [0.25 0.25 0.25 0.25]

Probability of each action in Cell(0 , 2) are: [0. 0. 0. 1.]

Probability of each action in Cell(0 , 3) are: [0.25 0.25 0.25 0.25]

Probability of each action in Cell(0 , 4) are: [0. 0. 0. 1.]

Probability of each action in Cell(1 , 0) are: [0.5 0.5 0. 0.]

Probability of each action in Cell(1 , 1) are: [1. 0. 0. 0.]

Probability of each action in Cell(1 , 2) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(1 , 3) are: [0. 0. 0. 1.]

Probability of each action in Cell(1 , 4) are: [0. 0. 0. 1.]

Probability of each action in Cell(2 , 0) are: [0.5 0.5 0. 0.]

Probability of each action in Cell(2 , 1) are: [1. 0. 0. 0.]

Probability of each action in Cell(2 , 2) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(2 , 3) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(2 , 4) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(3 , 0) are: [0.5 0.5 0. 0.]

Probability of each action in Cell(3 , 1) are: [1. 0. 0. 0.]

Probability of each action in Cell(3 , 2) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(3 , 3) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(3 , 4) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(4 , 0) are: [0.5 0.5 0. 0.]

Probability of each action in Cell(4 , 1) are: [1. 0. 0. 0.]

Probability of each action in Cell(4 , 2) are: [0.5 0. 0. 0.5]

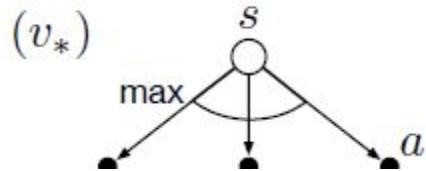
Probability of each action in Cell(4 , 3) are: [0.5 0. 0. 0.5]

Probability of each action in Cell(4 , 4) are: [0.5 0. 0. 0.5]

Question 5

Given an equation for v^* in terms of q^* .

$$v^*(s) = \max_a \{q^*(s, a)\}$$



From the backup diagram, we can see that to find the optimal state values function we have to take a single action and then find the maximum from all the q^* values.

Question 6

Implementation of the Policy Iteration and Value Iteration can be found in Question6_PolicyIteration.ipynb and Question6_ValuesIteration.ipynb respectively. Sample values and policy is printed there.

State Value of Each iteration

Policy Evaluation : Iteration 1	
1) [[0. -1. -1. -1.] [-1. -2. -2. -2.] [-1. -2. -2. -2.] [-1. -2. -2. 0.]]	2) [[0. -7. -10. -11.] [-7. -10. -11. -11.] [-10. -11. -10. -8.] [-11. -11. -8. 0.]]
3) [[0. -13. -19. -21.] [-13. -17. -19. -19.] [-19. -19. -17. -13.] [-21. -19. -13. 0.]]	4) [[0. -14. -20. -22.] [-14. -18. -20. -20.] [-20. -20. -18. -14.] [-22. -20. -14. 0.]]
5) [[0. -14. -20. -22.] [-14. -18. -20. -20.] [-20. -20. -18. -14.] [-22. -20. -14. 0.]]	

Policy Evaluation : Iteration 2

1) [[0. -1. -2. -3.] [-1. -2. -3. -15.] [-2. -3. -15. -1.] [-3. -15. -1. 0.]]	2) [[0. -1. -2. -3.] [-1. -2. -3. -2.] [-2. -3. -2. -1.] [-3. -2. -1. 0.]]
---	--

Policy Evaluation : Iteration 3

1) [[0. -1. -2. -3.] [-1. -2. -3. -2.] [-2. -3. -2. -1.] [-3. -2. -1. 0.]]
--

Policy Improvement : Iteration 1

state [Up, down, left, right] 1 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 2 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 3 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 4 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 5 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 6 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 7 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 8 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 9 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 10 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 11 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 12 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 13 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)] 14 [(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)]

Policy Improvement : Iteration 2

state [Up, down, left, right] 1 [(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)] 2 [(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)] 3 [(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)] 4 [(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)] 5 [(1, 0.5), (2, 0.0), (3, 0.5), (4, 0.0)] 6 [(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)] 7 [(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)] 8 [(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)] 9 [(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)] 10 [(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)] 11 [(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)]

12	$[(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)]$
13	$[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]$
14	$[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]$

Policy Improvement : Iteration 3

state	$[Up, down, left, right]$
1	$[(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)]$
2	$[(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)]$
3	$[(1, 0.0), (2, 0.5), (3, 0.5), (4, 0.0)]$
4	$[(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)]$
5	$[(1, 0.5), (2, 0.0), (3, 0.5), (4, 0.0)]$
6	$[(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)]$
7	$[(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)]$
8	$[(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)]$
9	$[(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)]$
10	$[(1, 0.0), (2, 0.5), (3, 0.0), (4, 0.5)]$
11	$[(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)]$
12	$[(1, 0.5), (2, 0.0), (3, 0.0), (4, 0.5)]$
13	$[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]$
14	$[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]$

Policy Improvement : Iteration 4

state	$[Up, down, left, right]$
1	$[(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)]$
2	$[(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)]$
3	$[(1, 0.0), (2, 0.5), (3, 0.5), (4, 0.0)]$
4	$[(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)]$
5	$[(1, 0.5), (2, 0.0), (3, 0.5), (4, 0.0)]$
6	$[(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)]$
7	$[(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)]$
8	$[(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)]$
9	$[(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)]$
10	$[(1, 0.0), (2, 0.5), (3, 0.0), (4, 0.5)]$
11	$[(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)]$
12	$[(1, 0.5), (2, 0.0), (3, 0.0), (4, 0.5)]$
13	$[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]$
14	$[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]$

Values Iteration's Iteration

1) $[[0. -1. -1. -1.]$ $[-1. -1. -1. -1.]$ $[-1. -1. -1. -1.]$ $[-1. -1. -1. 0.]]$	2) $[[0. -1. -2. -2.]$ $[-1. -2. -2. -2.]$ $[-2. -2. -2. -1.]$ $[-2. -2. -1. 0.]]$
--	--

3) [[0. -1. -2. -3.]
[-1. -2. -3. -2.]
[-2. -3. -2. -1.]
[-3. -2. -1. 0.]]

Value Iteration Policy

state	[Up, down, left, right]
1	[(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)]
2	[(1, 0.0), (2, 0.0), (3, 1.0), (4, 0.0)]
3	[(1, 0.0), (2, 0.5), (3, 0.5), (4, 0.0)]
4	[(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)]
5	[(1, 0.5), (2, 0.0), (3, 0.5), (4, 0.0)]
6	[(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)]
7	[(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)]
8	[(1, 1.0), (2, 0.0), (3, 0.0), (4, 0.0)]
9	[(1, 0.25), (2, 0.25), (3, 0.25), (4, 0.25)]
10	[(1, 0.0), (2, 0.5), (3, 0.0), (4, 0.5)]
11	[(1, 0.0), (2, 1.0), (3, 0.0), (4, 0.0)]
12	[(1, 0.5), (2, 0.0), (3, 0.0), (4, 0.5)]
13	[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]
14	[(1, 0.0), (2, 0.0), (3, 0.0), (4, 1.0)]

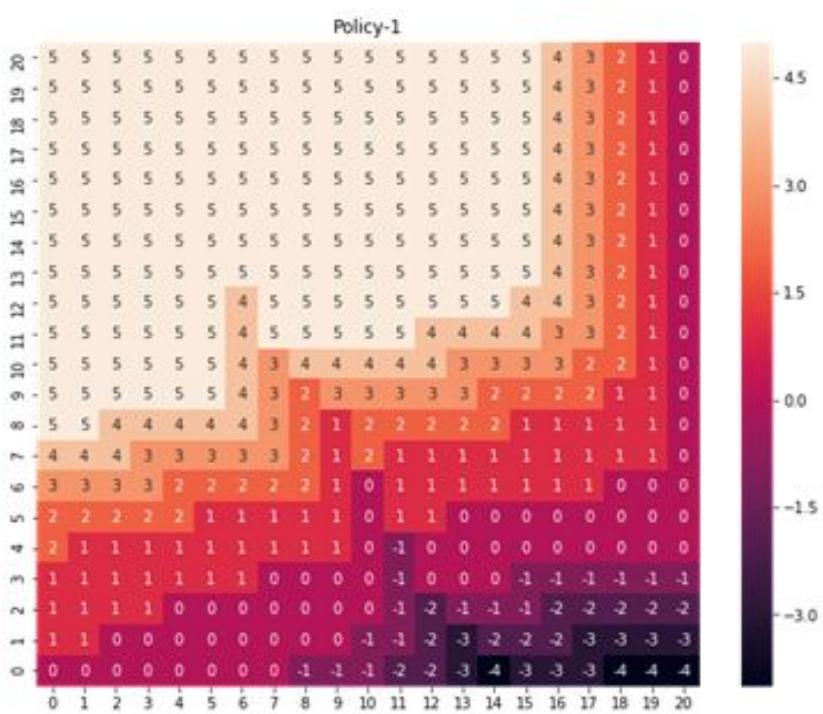
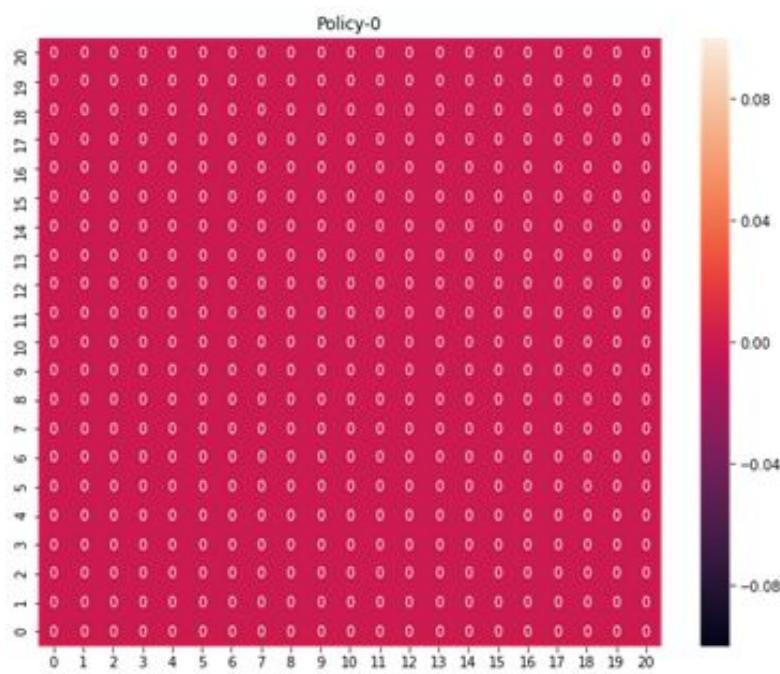
Question 7

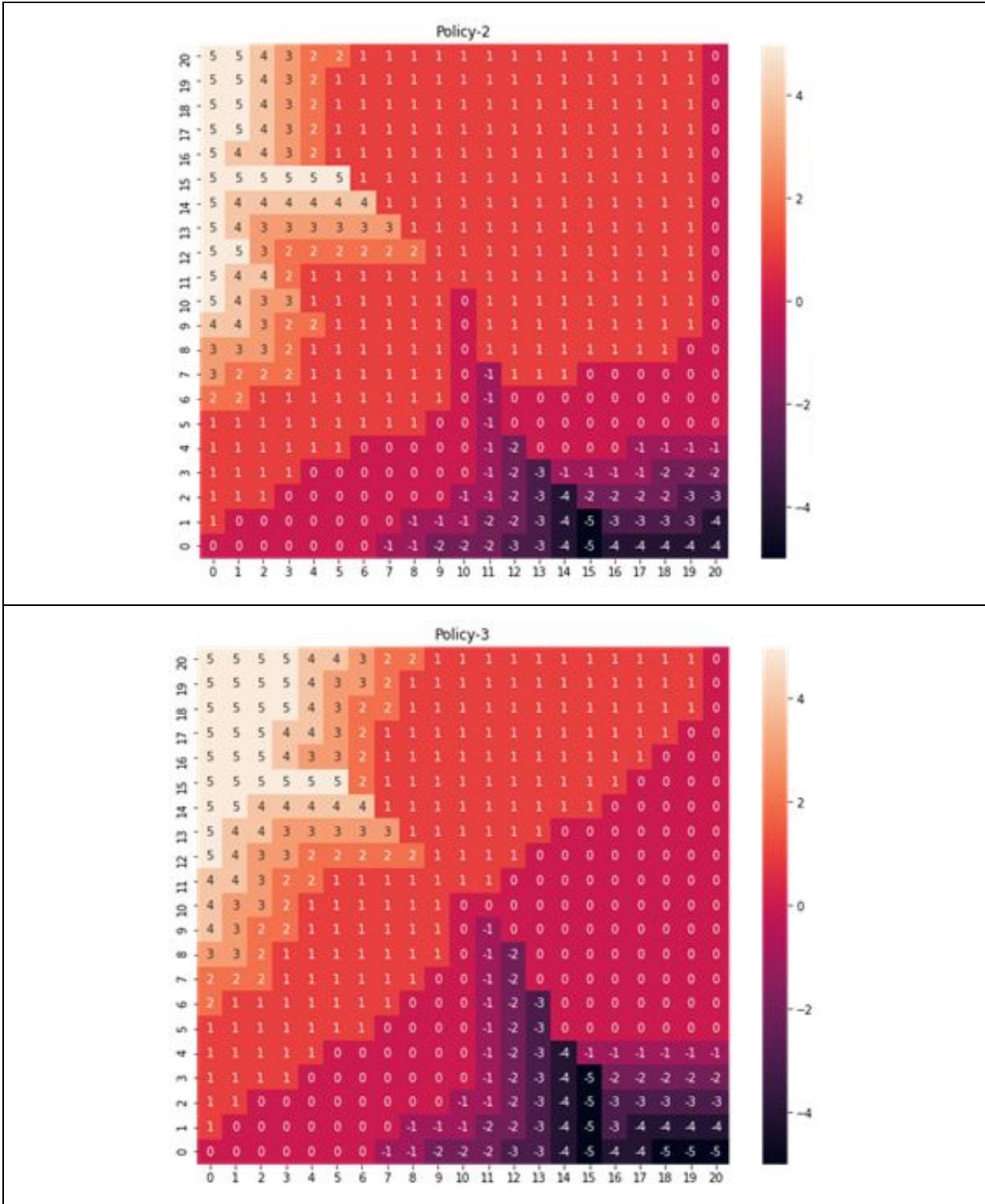
Policy Iteration for Jack Rental Car Problem.

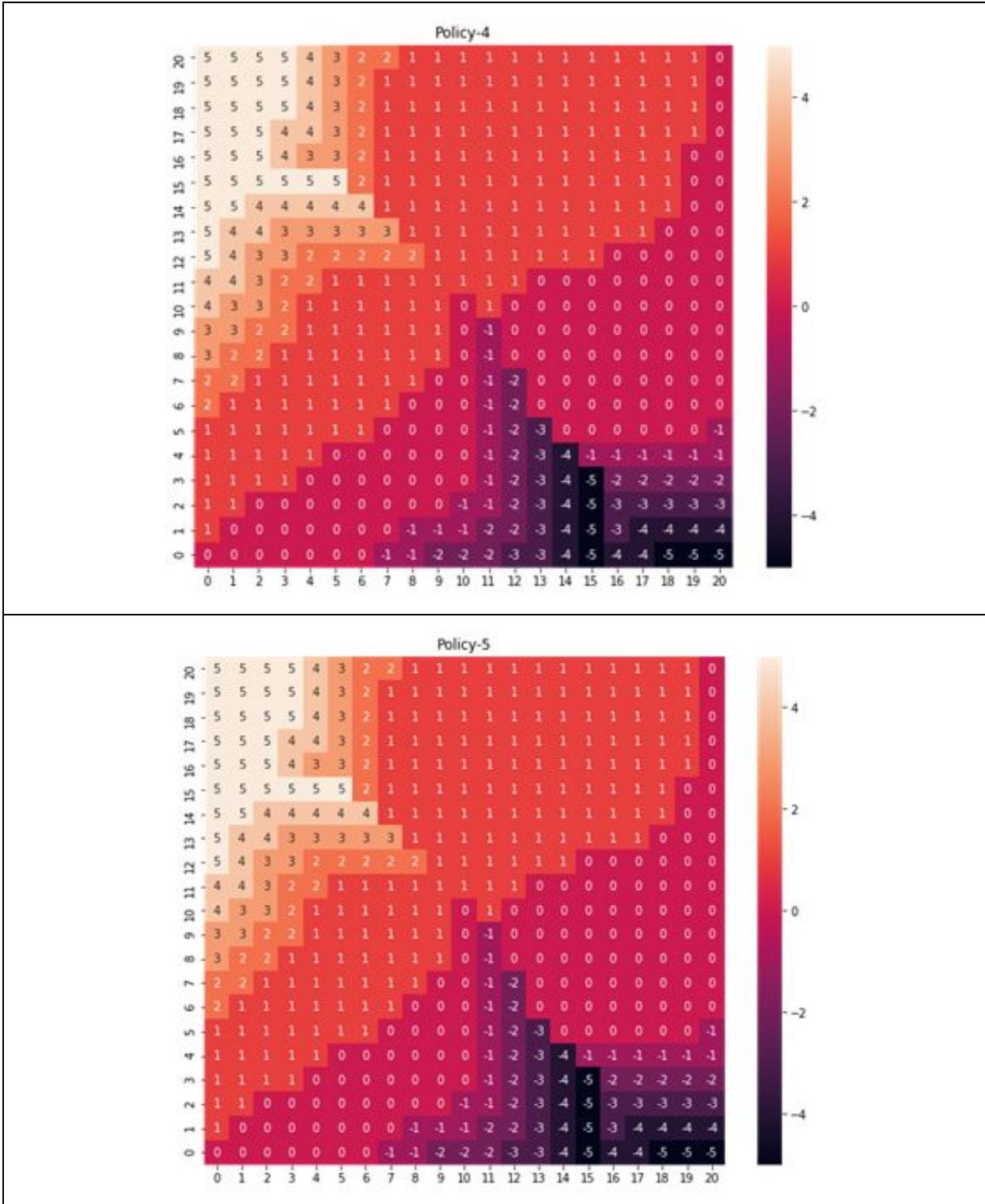
Assumption: While calculation the next state and reward we have taken a constant return of 3 at location A and 2 at location B.

For detail describtion see Question7_JackRental.ipynb.

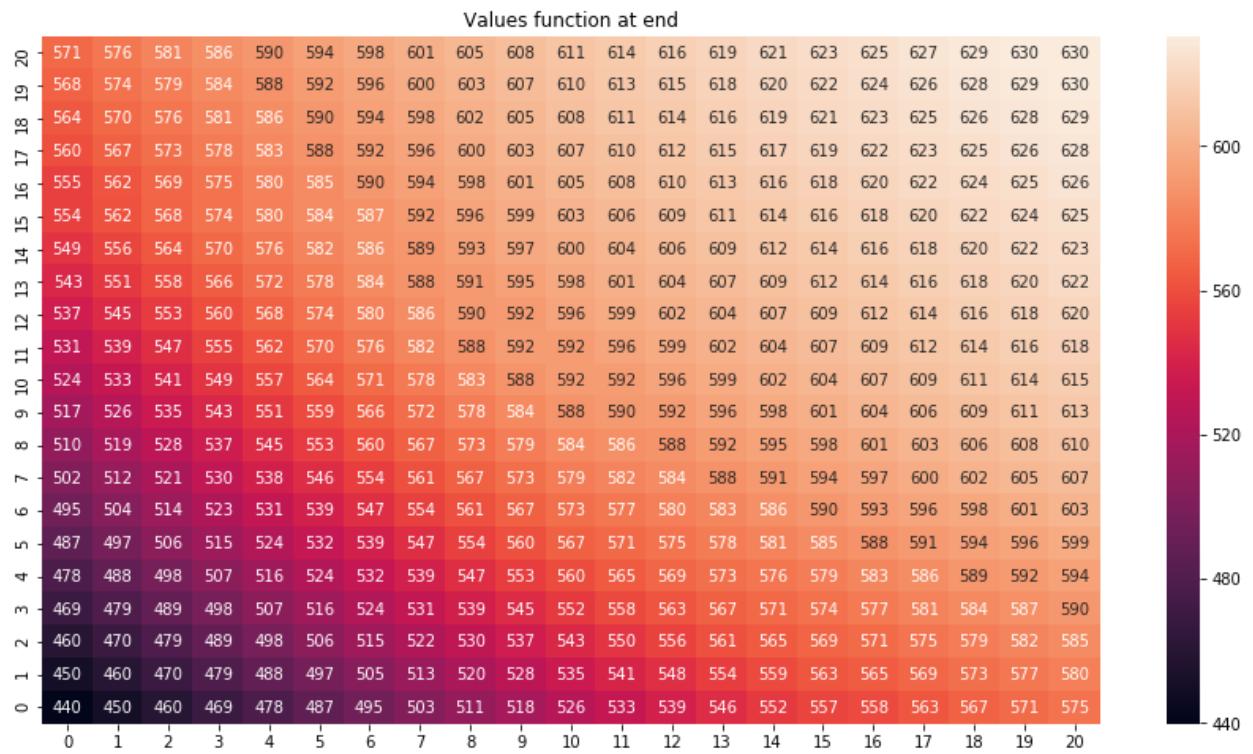
Below are the policy for every iteration of Policy Iteration







Value function after all the iteration:



Question 8

Q8

We have given a MDP
and we have to find that
if current state & action affect the
reward after the next state.

$$P.C R_{t+2} = r / S_t = s, A_t = a)$$

$$= P(R_{t+2} = r, S_t = s, A_t = a) \\ P(S_t = s, A_t = a)$$

Conditional probability can be written
as joint probability of Event divide
by individual probability.

Similarly

$$P(R_{t+2} = r / S_t = s, A_t = a, S_{t+1} = s', A_{t+1} = a')$$

$$= P(R_{t+2} = r, S_t = s, A_t = a / S_{t+1} = s', A_{t+1} = a')$$

$$P(S_t = s, A_t = a / S_{t+1} = s', A_{t+1} = a')$$

By Markov Property Property we know
that given the present state
& action, future is independent of
the past.

$$= P(R_{t+2} = r / S_{t+1} = s', A_{t+1} = a') \times$$

$$P(S_t = s, A_t = a / S_{t+1} = s', A_{t+1} = a')$$

$$P(S_t = s, A_t = a / S_{t+1} = s', A_{t+1} = a')$$



$$= P(R_{t+1} = r \mid S_{t+1} = s^1, A_{t+1} = a)$$

Page No.		
Date	/	/

So we can see given the current state (S_t) then R_{t+2} and (S_t, A_t) are conditionally independent.



Scanned with
CamScanner

Question 9

Q9. Derive a Expression for $E[R_{t+2} | S_t = s, A_t = a]$
in terms of PMF

$$E[R_{t+2} | S_t = s, A_t = a]$$

$$= \sum_{r \in R} r * p(R_{t+2} = r | S_t = s, A_t = a)$$

$$= \sum_{r \in R} r * \sum_{s' \in S} p(R_{t+2} = r | S_t = s, A_t = a, S_{t+1} = s', A_{t+1} = a')$$

if we ~~only~~ marginalize over S_{t+1} &
 A_{t+1} we get the same equation
as above.

In the summation we have not included
 A_{t+1} because we assume a
deterministic policy.

$$E[R_{t+2} | S_t = s, A_t = a] = \sum_{r \in R} r * \sum_{s' \in S} p(R_{t+2} = r | S_{t+1} = s', A_{t+1} = a')$$

$$= \sum_{r \in R} r * \sum_{s' \in S} \sum_{s'' \in S} p(s_{t+2} = s'', R_{t+2} = r | S_{t+1} = s', A_{t+2} = a')$$



Question 10

$$\text{Ans} \rightarrow V_{\pi}(s) = E_{\pi}[G_t | S_t = s]$$

$$V_{\pi}(s) = E_{\pi}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \dots | S_t = s]$$

$$= E_{\pi}[R_{t+1} + \gamma (R_{t+2} + \gamma R_{t+3} + \gamma^2 R_{t+4}) | S_t = s]$$

$$= E_{\pi}[R_{t+1} + \gamma G_{t+1} | S_t = s]$$

, because $G_{t+1} = R_{t+2} + \gamma R_{t+3} + \gamma^2 R_{t+4}$

Now as expectation is over policy π
we will open the expectation

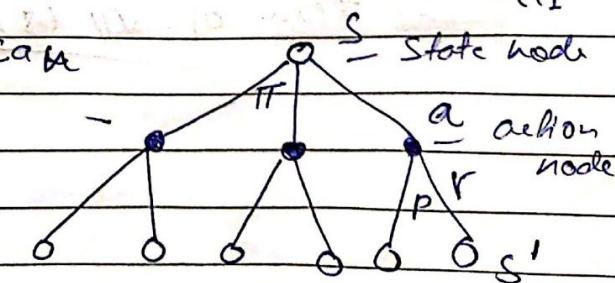
$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r | s, a) [R + \gamma E_{\pi}[G_{t+1} | S_{t+1} = s']]$$

The above expansion can

be understood with

the help of this

back up diagram



From a state with respect to some particular policy we

choose a action that will take us to action node. From action node environment can blow me to any of the new value state s' . So for computing we have to consider all possibility.

$$V_{\pi}(s) = \sum_a \pi(a|s) \sum_{s', r} p(s', r|s, a) [r + \gamma E_{\pi}[G_{t+1} | S_{t+1} = s']]$$

By definition definition of V_{π} we know

$$V_{\pi}(s') = E_{\pi}[G_t | S_t = s']$$

so above equation becomes

$$V_{\pi}(s) = \sum_a \pi(a|s) \sum_{s', r} p(s', r|s, a) [r + \gamma V_{\pi}(s')]$$

for all $s \in S$

This is Bellman equation of $V_{\pi}(s)$ in term of $V_{\pi}(s)$.

Question 11

Q11

$$R_1 = 2, R_2 = -1, R_3 = 10$$

$$R_4 = -3$$

Page No.

Date

We have to calculate discounted return for each time step.

$$G_t = R_{t+1} + \gamma G_{t+1}$$

Return

for first let start from $t=4$

$$G_4 = R_5 + \gamma G_5$$

$$= 0 + 0 = 0$$

$$\text{for } t=3 \quad G_3 = R_4 + \gamma G_4$$

$$= -3 + 0.5 \times 0$$

$$= -3$$

$$\text{for } t=2 \quad G_2 = R_3 + \gamma G_3$$

$$= 10 + 0.5 \times -3$$

$$= 10 + -1.5 = 8.5$$

$$\text{for } t=1 \quad G_1 = R_2 + \gamma G_2$$

$$= -1 + 0.5 \times 8.5$$

$$= -1 + 4.25 = 3.25$$

$$\text{for } t=0 \quad G_0 = R_1 + \gamma G_1$$

$$= 2 + 0.5 \times 3.25$$

$$= 2 + 1.625 = 3.625$$

Now suppose if we get a constant reward a then return



Scanned with
CamScanner

$$C_{t+3} = R_{t+1} + V R_{t+2} + V^2 R_{t+3}$$

Page No. _____

Date _____

$$= C + V C + V^2 C + \dots$$

$$= C [1 + V + V^2 + \dots]$$

- This forms a GP (Geometric Progression).

Sum of AP when common ratio is less than
1 as here ($V < 1$) = $\frac{a}{1 - r}$

Here $a = 1$, $r = V$

$$\text{So } C \times \frac{1}{1 - V} = \frac{C}{1 - V}$$



Scanned with
CamScanner

Question 12

12

~~If we have been given optimal state-value function $V_\pi(s)$.~~

~~& we have to find optimal policy.~~

Optimal policy is from a state which action we should pick so that our expected reward is maximized -

Optimal State-Value function is ~~the~~ expect value one can achieve from a state following a ~~the~~ Optimal policy state

Now when Optimal Value is given finding optimal policy is like a one step look ahead -



Scanned with
CamScanner

So from a state take all action and see which &

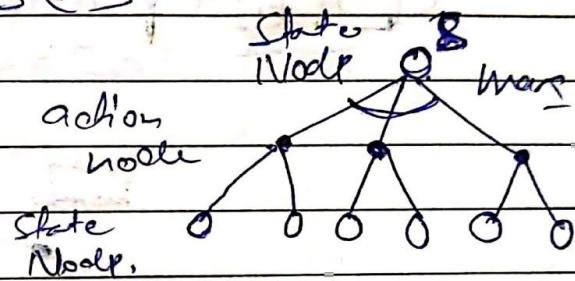
Page No.

Date

action is giving maximum return.

$$a = \arg \max_a \sum_{s', r} p(s', r | s, a) [r + V_\pi(s')]$$

for all $s \in S$



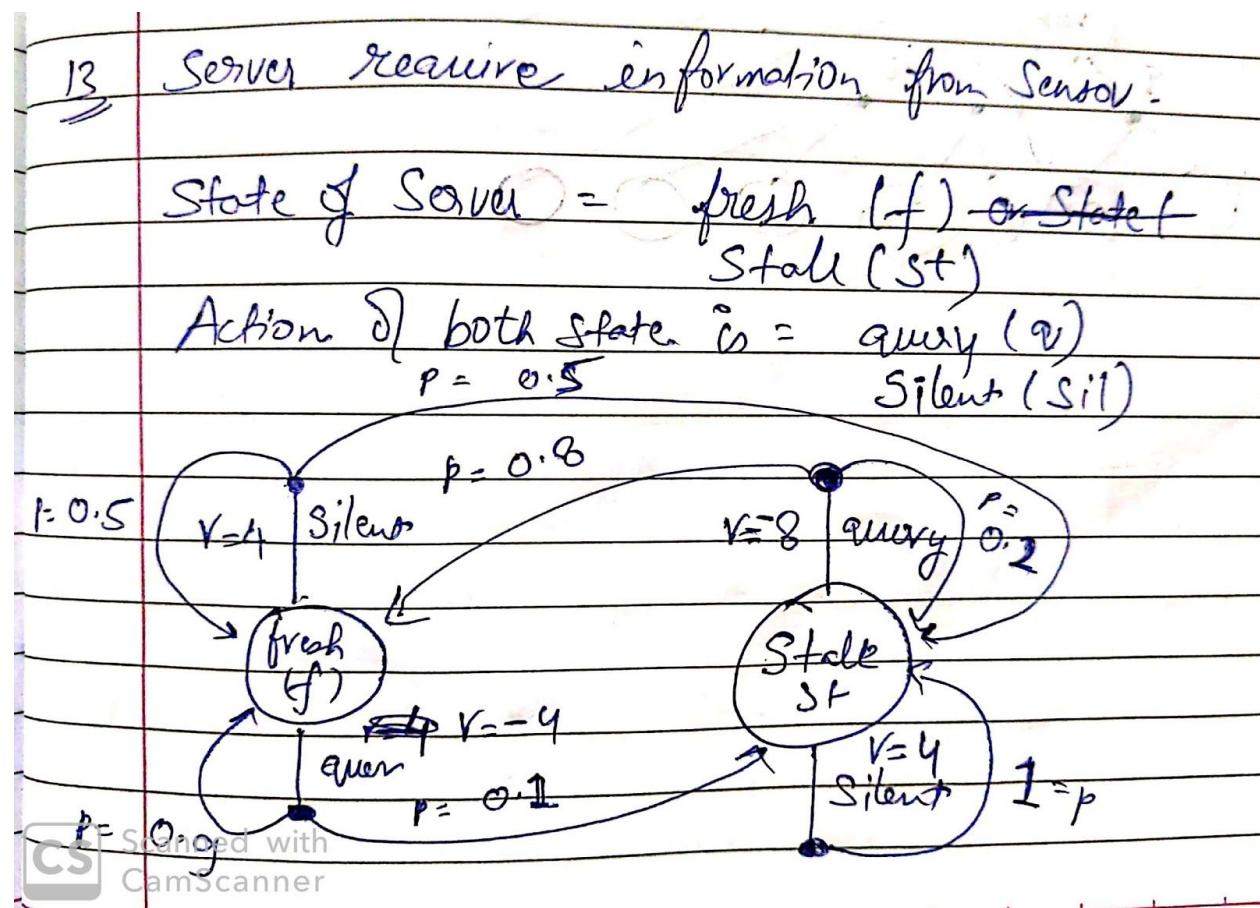
→ From Node S take all action and do one step look ahead & see that reward is getting

→ Then optimal action is the action that yield maximum reward



Scanned with
CamScanner

Question 13 (a)



Question 13 (b)

13

Finite horizon Problem
Optimize cost over three time step

Page No.

Date

- At the end we get +10 ~~cost~~ reward if we are at fresh state -10 otherwise
- Discount factor = 0.5
- ~~we have to find optimal cost~~ reward

To find optimal ~~cost~~ we can explore every path till 3 time step and then calculate the ~~cost~~ reward

$$V_x(s) = \max_a q(s, a)$$

So we can take every action from state and see which gives us the best return.

Assumption

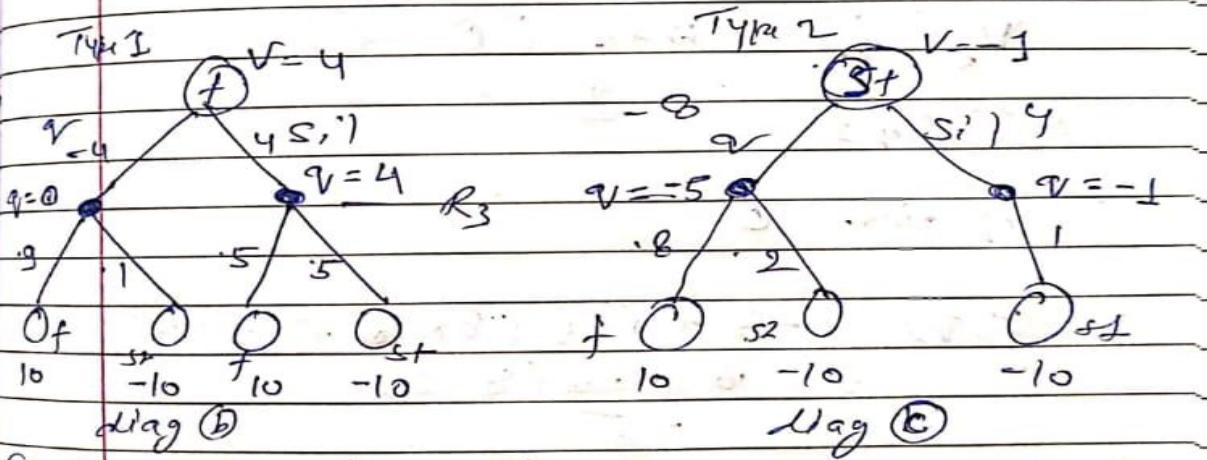
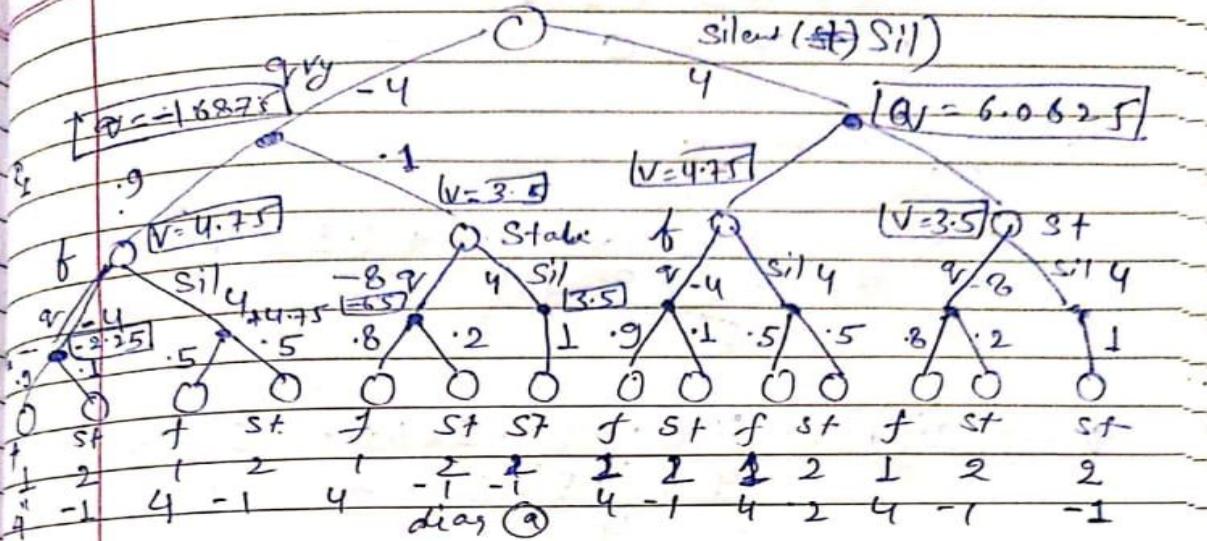
- ① 0 - represent ~~state~~ node. \bullet Repres action node
- ② Value over from ~~state~~ state node to action node is reward
- ③ Value over action node to state node represents probability.
- ④ The diagram shows ~~back~~ space for two time step.
- ⑤ Tree diagram for 3rd is shown ~~separately~~ separately.



Page No. _____
Date _____

$$V_f = 6.0625$$

fresh



- (b) After the two time step we get a reward to 10 so if we are in fresh so we have taken value 10 at Time 3rd time step likewise.

- (c) for calculating the α value

$$V(s, a) = R + \gamma \sum_{s'} p(s'|s, a) (\alpha V(s'))$$

Solving diag b first

$$q(0,0) = -4 + 0.5(0.9 \times 10 + 1.8 \times -10)$$
$$= -4 + 0.5(9 - 18)$$
$$= -4 + 4 = 0$$

Page No.

Date

Solving diagram c

$$= -8 + 0.5((0.8 \times 10) + (-2 \times 10))$$
$$= -8 + 0.5(8 - 2)$$
$$= -5$$

$$q(f, \text{sil}) = 4 + 0.5(0.5 \times 10 + 0.5 \times -10)$$
$$= 4 + 0.5(5 - 5)$$
$$= 4$$

$$v_s(f) = \max(0, 4)$$
$$= 4$$

At Time Step 3

Solving diag c

$$q(f, \text{st, q}) = -8 + 0.5((0.8 \times 10) + (-2 \times 10))$$
$$= -8 + 0.5(8 - 2)$$
$$= -5$$

$$q(f, \text{st, sil}) = 4 + 0.5(1 \times -10)$$
$$= 4 - 5 = -1$$

$$v_s(s) = \max(-5, -1)$$
$$= -1$$

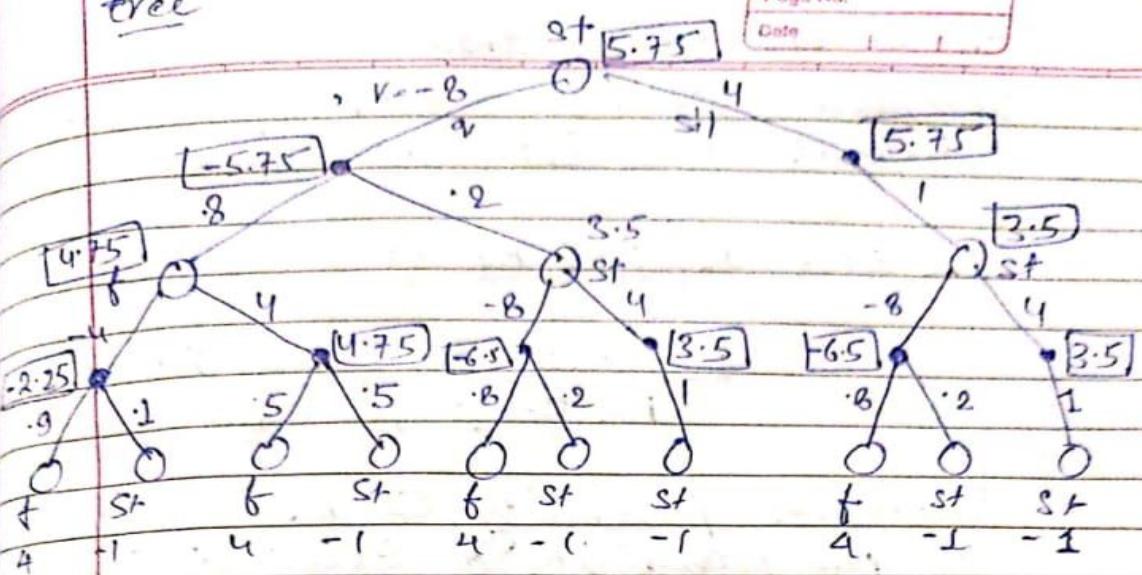
So At Value of node at the end for diag a

is 4 & -1.



Scanned with
CamScanner

like wise I have filled the complete tree



$$\text{so } V^*(\text{fresh}) = 6.0625 \quad \text{optimal value}$$

$$V^*(\text{stale}) = 5.75 \quad \text{Value}$$

optimal policy $\pi^*(s) = \arg \max_a q^*(s, a)$

$$\pi(a/\text{fresh}) = \cancel{\text{Silent}}$$

$$\pi(a/\text{stale}) = \cancel{\text{Silent}}$$

$$\pi(a/\text{fresh}) = \arg \max_{q_{ry}} (-16.875, 6.0625)$$

= Silent

$$\pi(a/\text{stale}) = \arg \max_{q_{ry}} (-5.75, 5.75)$$

= Silent

Question 13 (c)

B) Infinite horizon setting.

$$\text{Initialize } V(f) = 0 \quad V(s_f) = 0$$

Iteration 1

$$\begin{aligned} V(f) &= \max \left(0.9(-4 + 0.5 \times V(f)) + 0.1(-4 + 0.5 \times V(s_f)), \right. \\ &\quad \left. 0.5(4 + 0.5 \times V(f)) + 0.5(4 + 0.5 \times V(s_f)) \right) \\ &= \max \left(-0.9 \times -4 + 0.1 \times -4, \right. \\ &\quad \left. 0.5 \times 4 + 0.5 \times 4 \right) \\ &= \max \left(\frac{-3.6 - 4}{2+2} \right) = \max \left(\frac{-4}{4} \right) = 4 \end{aligned}$$

$$\begin{aligned} V(s_f) &= \max \left(0.8(-8 + 0.5 V(f)) + 0.2(-8 + 0.5 V(s_f)), \right. \\ &\quad \left. 1 \times (4 + 0.5 V(s)) \right) \\ &= \max \left(\frac{-6.4 - 1.6}{2+1} \right) = 4 \end{aligned}$$



Scanned with
CamScanner

Iteration 2

$$V(f) = \max \left(0.9(-4 + 0.5V(f)) + 0.1(-4 + 0.5V(s)) \right)$$
$$\quad \cdot 5(4 + 0.5V(f)) + 0.5(4 + 0.5V(s)) \right)$$

$$\max \left(0.9(-4 + 0.5 \times 4) + 0.1(-4 + 0.5 \times 4) \right)$$
$$\cdot 5(4 + 0.5 \times 4) + 0.5(4 + 0.5 \times 4) \right)$$

$$\max = \begin{pmatrix} 0.9 \times -2 & 0.1 \times -2 \\ 0.5 \times 6 & 0.5 \times 6 \end{pmatrix}$$
$$= \begin{pmatrix} -2 \\ 6 \end{pmatrix} = 6$$

$$V(s) = \max \left(0.8(-8 + 0.5 \times 4) + 0.2(-8 + 0.5 \times 4) \right)$$
$$\cdot 1(4 + 0.5 \times 4) \right)$$

$$\Rightarrow \begin{pmatrix} 0.8(-6) + 0.2 \times -6 \\ 4 + 2 \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} -8 \\ 6 \end{pmatrix} = 6$$

Iteration 3

$$V(f) = \max \left(0.9(-4 + 0.5 \times 6) + 0.1(-4 + 0.5 \times 6) \right)$$
$$\cdot 5(4 + 0.5 \times 6) + 0.5(4 + 0.5 \times 6) \right)$$

$$\max = \begin{pmatrix} -1 \\ 7 \end{pmatrix} = 7$$

$$V(s) = \max \left(0.8(-8 + 0.5 \times 6) + 0.2(-8 + 0.5 \times 6) \right)$$
$$\cdot 1(4 + 0.5 \times 6) \right)$$

$$\Rightarrow \begin{pmatrix} -5 \\ 7 \end{pmatrix} = 7$$

Iteration 4

$$V(f) = \max \left(0.9(-4 + 0.5 \times 7) + 0.1(-4 + 0.5 \times 7) \right)$$
$$\cdot 5(4 + 0.5 \times 7) + 0.5(4 + 0.5 \times 7) \right)$$

$$V(s) = \max \left(\begin{array}{l} 0.8(-8 + 0.5 \times 7) + 0.2(-8 + 0.5 \times 7) \\ 1(4 + 0.5 \times 7) \end{array} \right)$$

Page No. _____
Date _____

$$= \max \left(\begin{array}{l} -4.5 \\ 7.5 \end{array} \right) = \underline{\underline{7.5}}$$

Policy After Converge is

State	Action	
	Query	Silent
1	0.0	1.0
2	0.0	1.0

Scanned with
CamScanner

13.3 Policy Iteration

Initializing

$$V(f) = 0 \quad V(sf) = 0$$

Policy after init: equiprobable policy

$$\pi(\text{quen}/f) = 0.5 \quad \pi(\text{quen}/sf) = 0.5$$

$$\pi(\text{silent}/f) = 0.5 \quad \pi(\text{silent}/sf) = 0.5$$

Iteration 1

$$\text{Policy Evaluation} = V(s) = \sum_{s'} p(s', r | s, \pi(s)) [r + V(s')]$$

Policy Evaluation - Iteration 1

$$V(f) = 0.5 [0.9(-4 + 0.5 * V(f)) + 0.1(-4 + 0.5 * V(sf))]$$

$$+ 0.5 [0.5(4 + 0.5 * V(f)) + 0.5(4 + 0.5 * V(sf))]$$

$$= 0.5 [0.9(-4) + 0.1(-4)] + 0.5 [0.5(4) + 0.5(4)]$$

$$\Rightarrow -2 + 2 = 0$$

$$V(sf) = 0.5 [0.8(-8 + 0.5 * V(f)) + 0.2(-8 + 0.5 * V(sf))]$$

$$+ 0.5 [4 + 0.5 * V(sf)]$$



Scanned with
CamScanner

$$V(S_t) = 0.5 [0.8(-8) + 0.2(-8)] \\ \rightarrow 0.5 [4]$$

Page No.	1
Date	

$$\Rightarrow 0.5(-8) + 0.5(4) \\ -4 + 2 = -2$$

Policy Evaluation Iteration 2

$$V(f) = 0.5 [0.9(-4+0) + 0.1(-4+0.5(-2))]$$

$$+ 0.5 [0.5(4+0.5*0) + 0.5(4+0.5(-2))]$$

$$0.5[0.9(-4) + 0.1(-5)] \\ + 0.5[2 + 1.5] \\ \rightarrow 0.5[-3.6 - 0.5] + 0.5(3.5) \\ \Rightarrow -2.05 + 1.75 \\ = \cancel{-0.3} - 0.3$$

$$V(S_t) = 0.5 [0.8(-8 + 0.5(-0.3)) + 0.2(-8 + 0.5(-2))] \\ + 0.5(4 + 0.5(-2))$$

$$\Rightarrow 0.5 [0.8(-8 - 0.15) + 0.2(-8 - 1)] + \\ 0.5(4 - 1)$$

$$\Rightarrow 0.5 [-6.52 - 1.8] + 1.5$$

$$\Rightarrow \cancel{-4.16} + 1.5 - 4.16 + 1.5 \\ - \cancel{2.52} - 2.66$$

Policy Evaluation Iteration 3

$$V(f) = 0.5 [0.9(-9 + 0.5(-0.3)) + 0.1(-9 + 0.5(-2.66))] \\ + 0.5(4 + 0.5(-0.3))$$

$$+ 0.5 [0.5[4 + 0.5(-0.3)] + 0.5(4 + 0.5(-2.66))]$$

$$\begin{aligned}
 &= 0.5[-3.735 + -533] + \\
 &= 0.5[1.925 + 1.335] \\
 \Rightarrow &= -2.134 + 1.65 \\
 &= -0.504
 \end{aligned}$$

Page No. _____
Date _____

$$\begin{aligned}
 v(st) &= 0.5[0.8(-8 + 0.5(-0.504)) + 0.2(-8 + 0.5(-2.89)) \\
 &\quad + 0.5[4 + 0.5(-2.66))] \\
 &= 0.5[-6.6016 + -1.866] \\
 &\quad + 0.5[4 - 1.33] \\
 &= -4.2338 + 1.335 \\
 &= -2.8988
 \end{aligned}$$

Policy Improvement.

$$\pi(a/f) = \arg \max_a \left(0.9(-4 + 0.5 \times (-0.504)) + 0.1(-4 + 0.5 \times -2.89) \right. \\
 \left. + 0.5(4 + 0.5 \times (-0.504)) + 0.5(4 + 0.5 \times -2.89) \right)$$

$$\begin{aligned}
 &\stackrel{\text{avg}}{\max} \left(0.9(-4 + -0.252) + 0.1(-4 + -1.445) \right) \\
 &\quad \left. + 0.5(4 + -0.252) + 0.5(4 + -1.445) \right)
 \end{aligned}$$

= action 2 is ~~worse~~ i.e., Silent.

$$\pi(a/st) = \arg \max_a \left(0.8(-8 + 0.5 \times (-0.504)) + 0.2(-8 + 0.5 \times (-2.89)) \right. \\
 \left. + 4 + 0.5 \times (-2.66) \right)$$

Action 2 ie Silent is better.

$$\pi(a/f) - \pi(a/st) = 2 (Silent)$$

Question 14

Question 15

Question 15 Stairwell

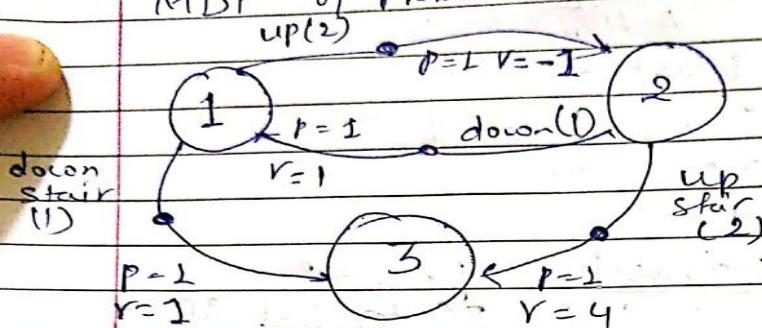
Page No.

Date

States = 1, 2, 3 (terminal)

Action from each state = { down-stair (1)
except terminal up-stair (2) }

MDP Problem



Use Policy Iteration to find optimal policy for
n=2

Policy Iteration has two step policy evaluation & policy improvement

Initializing

$$V = [0, 0, 0]^T$$

$$V(1) = 0$$

$$V(2) = 0$$

$$V(3) = 0$$

Policy is a equiprobable policy ($P(a|s)$)

$$P(1|1) = 0.5$$

$$P(1|2) = 0.5$$

$$P(2|1) = 0.5$$

$$P(2|2) = 0.5$$

Evaluation 1

PEI Policy Evaluation $V(s) = \sum_{s', v} p(s', v | s, \pi(s)) [v + \gamma V(s')]$

$$V(1) = 0.5 \left(1(1 + 1 * V(3)) \right) + 0.5 \left(1(-1 + 1 * V(2)) \right)$$

rewar, gamma, value of next state



Scanned with
CamScanner

$$V(1) = 0.5(1) + 0.5(-1)$$

$$= 0$$

Page No.	
Date	

$$V(2) = 0.5(1+0) + 0.5(4+0)$$

$$= 0.5 + 2 = 2.5$$

PE2 - Policy Evaluation Step 2

$$V(1) = 0.5(1+0) + 0.5(-1+2.5)$$
~~- 0.55~~ $= 0.5 + 0.75 = 1.25$

$$V(2) = 0.5(1+1.25) + 0.5(4)$$

$$\Rightarrow 0.5(2.25) + 2$$

$$\approx 3.125$$

Policy Evaluation Step 3

$$V(1) = 0.5(1+0) + 0.5(-1+3.125)$$

$$\Rightarrow 0.5 + 1.0625$$

$$\approx 1.5625$$

$$V(2) = 0.5(1+1.5625) + 0.5(4)$$

$$\approx 1.28125 + 2$$

$$\approx 3.28125$$

Policy Evaluation Step 4

$$V(1) = 0.5(1) + 0.5(-1+3.28125)$$

$$= 0.5 + 1.140625$$

$$1.640625$$

$$V(2) = 0.5(1+1.640625) + 0.5(4)$$

$$= 1.3203125 + 2$$
~~3.3203125~~



We are doing Policy Evaluation till 4th
Step only.

Page No.
Date

Policy Improvement

$$a = \arg \max_a \sum_{s, r} p(s, r | s, a) [r + \gamma V(s)]$$

$$V(1) = \arg \max \left(\begin{array}{c} -1 + V(3) \\ -1 + V(2) \end{array} \right)$$
$$\approx \left(\begin{array}{c} 1 \\ -1 + 3.3203125 \end{array} \right) \rightarrow \frac{2}{2} \cancel{3}$$

$$V(2) = \arg \max \left(\begin{array}{c} 1 + 1.640625 \\ \cancel{4} \end{array} \right) = \cancel{1} 2$$

$$\text{Policy } \pi(a|1) = 2 \quad \text{Pi}(a|2) = 1$$

Policy Evaluation of new Policy

PEL

$$V(1) = -1 + 3.3023125 \rightarrow \text{Now we are considering only option 2 as policy}$$

$$= 2.3023125$$

$$V(2) = 4$$

Policy Evaluation Iteration 2

$$V(1) = -1 + 4 = 3$$

$$V(2) = 4$$

Policy Evaluation Iteration 3

$$V(1) = -1 + 4 = 3$$

$$V(2) = 4$$



As & There is no improvement
in policy we can

Page No.

Date

do the policy Improvement Step

$$V(1) = \underset{a}{\operatorname{argmax}} \left(\frac{1+0}{-1+4} \right) = \underset{a}{\operatorname{argmax}} \left(\frac{1}{3} \right) = \text{action 2}$$

$$V(2) = \underset{a}{\operatorname{argmax}} \left(\frac{1+3}{4} \right) = \underset{a}{\operatorname{argmax}} \left(\frac{4}{4} \right) =$$

both action with 50% prob

$$\pi(2|1) = 1 \quad \pi(1|2) = 0.5 \quad \pi(2|2) = 0.5$$

Policy Evaluation with New Policy

$$V(1) = -1 + 4 = 3$$

$$V(2) = 0.5(1+3) + 0.5(4)$$

$$= 2 + 2 = 4$$

As There is no improvement in state value.

∴ State go even again Improve
policy

Policy Improvement Step

$$V(1) = \underset{a}{\operatorname{argmax}} \left(\frac{1+0}{-1+4} \right) = \underset{a}{\operatorname{argmax}} \left(\frac{1}{3} \right) = \text{action 2}$$

$$V(2) = \underset{a}{\operatorname{argmax}} \left(\frac{1+3}{4} \right) = \underset{a}{\operatorname{argmax}} \left(\frac{4}{4} \right) =$$

both action with 50% prob

& final Prob

State	1	2
1	0.0	1.0
2	0.5	0.5



Scanned with
CamScanner

