EE675A : Introduction to Reinforcement Learning

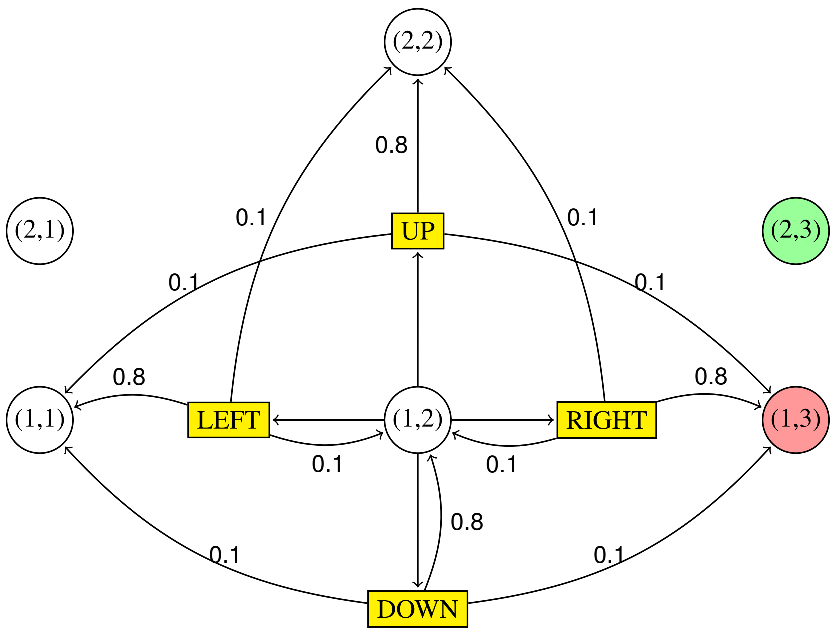
EE698V Mid-Semester Exam Solutions February 2022

03-05-2023

Lecturer: Subrahmanya Swamy Peruru *Scribe: Pranay Vandanapu*

1 Question 1

1.1 (a)



1.2 (b)

For $\gamma = 1$, the optimal policy is:

S	(1, 1)	(1, 2)	(1, 3)	(2, 1)	(2, 2)	(2, 3)
π^*	UP	LEFT	NA	RIGHT	RIGHT	NA

The optimal policy depends on γ.

For low values of γ, UP might be the best action in state (1, 2).

1.3 (c)

$V^*(s) = +5 \;\forall$ s $\notin \{(1,3),(2,3)\}$

$V_1(1,3) = V_1(2,3) = 0$. Since those are terminal states

1.4 (d)

$V_0(s) = 0$ ∀ s ∈ S.

S	(1, 1)	(1, 2)	(1, 3)	(2, 1)	(2, 2)	(2, 3)
V_1	0	0	0	0	4	0
V_2	0	2.38	0	2.88	4.36	0

$V_1(2, 2) = 0.8 * 5 = 4$

$V_2(1, 2) = 0.8(0 + 0.9 * 4) + 0.1(-5 + 0) = 2.38$

$V_2(2, 1) = 0.8(0 + 0.9 * 4) = 2.88$

$V_2(2, 2) = 0.8 * 5 + 0.1(0 + 0.9 * 4) = 4.36$

1.5 (e)

$V(1, 1) = (-5 + 5 + 5)/3 = 5/3$

$V(2, 2) = (5 + 5)/2 = 5$

1.6 (f)

$V(s) = V(s) + \alpha * (r + \gamma * V(s') - V(s))$

$\underline{Trail\ 1}$

$V(1, 2) = 0 + 0.1(-5 + 0.9 * 0 - 0) = -0.5$

No other updates.

$\underline{Trail\ 2}$

$V(1, 1) = 0 + 0.1(0 + 0.9 * -0.5 - 0) = -0.045$

$V(1, 2) = -0.5 + 0.1(0 + 0.9 * 0 + 0.5) = -0.45$

$V(2, 2) = 0 + 0.1(5 + 0.9 * 0 - 0) = 0.5$

2 Question 2

2.1 (a)

$G_t = R_{t+1} + \gamma * R_{t+2} + ... = \sum_{k=0}^{\infty} \gamma^k * R_{t+k+1}$

Adding c to each reward

$\tilde{G}_t = R_{t+1} + \gamma * R_{t+2} + ... + c + \gamma * c + ...$

$\tilde{G}_t = G_t + c * \sum_{k=0}^{\infty} \gamma^k$

$\tilde{G}_t = G_t + c/(1 - \gamma)$

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

$\tilde{V_\pi}(s) = E[\tilde{G_t}|S_t = s]$
$\tilde{V_\pi}(s) = E[G_t|S_t = s] + c/(1-\gamma)$
$\tilde{V_\pi}(s) = V_\pi(s) + V_c$
$\implies$ c doesn't affect the relative difference among states.

2.2 (b)

In episodic tasks adding a constant could change the goal.
For example in the shortest path grid problem, if we increase the reward of '-1' per step to '1' and terminal reward to '2', the agent will not reach the goal state.

3 Question 3

3.1 Dynamic programming

DP Backup Diagram

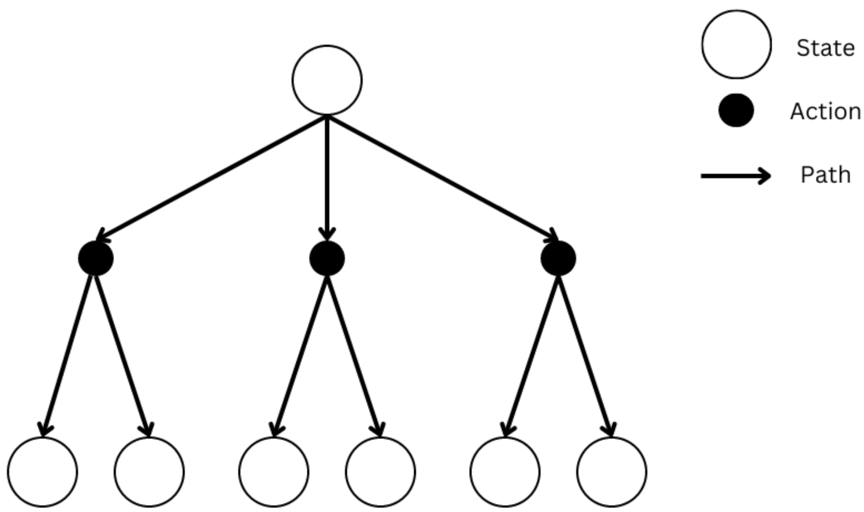


Figure 1: DP Backup Diagram

3.2 Monte-Carlo

Monte Carlo Backup Diagram

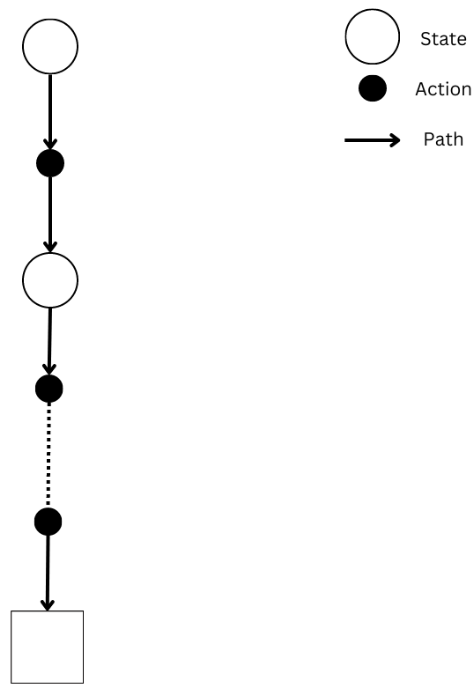


Figure 2: Monte Carlo Backup Diagram

3.3 TD

TD Backup Diagram

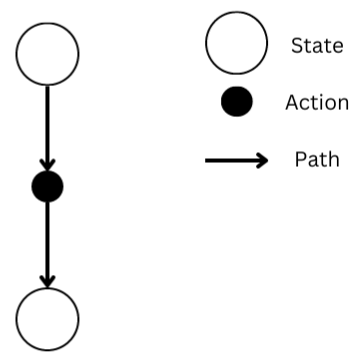


Figure 3: TD Backup Diagram

4 Question 4

4.1 (a)

States : 2^n states (Binary vector of length n), 1/0 - chair occupied or not.

Action : Picking a chair to occupy from a set of available chairs.

Transition : When picking a chair, the status of the chair goes from unoccupied to occupied.

Upon taking action.

Reward :

- $+1$ if nobody is sitting on the chair and adjacent chairs.
- -100 if only one of the neighbouring chairs is filled.
- -200 if both the neighbouring chairs are filled.

4.2 (b)

There are 2^6 states.

Only 18 of those states are valid:

- no occupied chair.

- 6 cases of 1 occupied chair.
- 9 cases of 2 occupied chairs.
- 2 cases of 3 occupied chairs.

Terminal states:

- Chairs 1, 3, and 5 are occupied.
- Chairs 2, 4, and 6 are occupied.
- Chairs 1 and 4 are occupied.
- Chairs 2 and 5 are occupied.
- Chairs 2 and 6 are occupied.

4.3 (c)

The given state is a terminal state.