

(s-silent, q-query)

Q 13. MDP.

1.

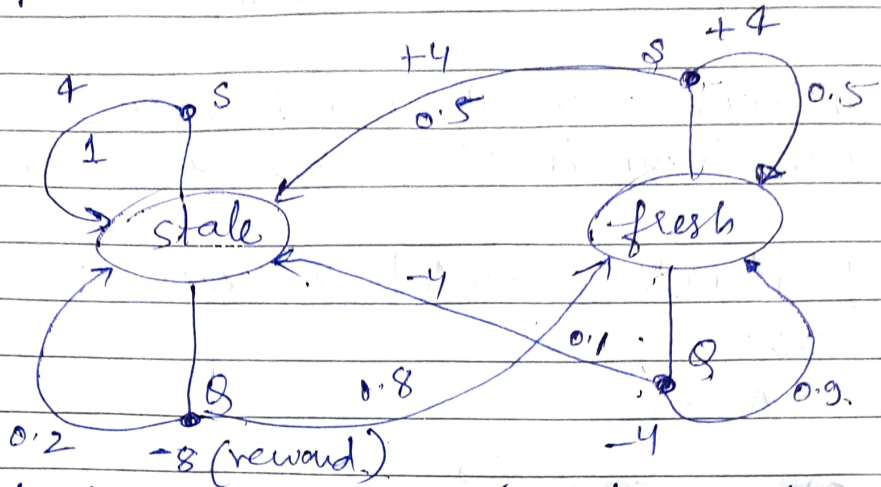


Table drawn in the jupyter coded file.

3. discount factor  $\gamma = 0.5$  ( $\gamma$ )

$S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3$   
at 3rd time step.

If information,  $R_3 = \begin{cases} -10, & S_2 = \text{stale} \\ +10, & S_2 = \text{fresh} \end{cases}$

Let  $S = \text{stale}$ ,  $F = \text{fresh}$

$V_{\pi}(\text{state}) =$  value functions.

$$\textcircled{A} \quad V_q(s) = \frac{1}{2} \left( 0.8 (-8 + 0.5 \cdot V_q^+(F)) \right) + \frac{1}{2} \left( 0.2 (-8 + V_q(s)) \right) + \frac{1}{2} (4 + 0.5 V_q(s)) \quad \times 1$$

$$\textcircled{B} \quad V_A(F) = \frac{1}{2} \left( 0.9 \left( -4 + \overset{0.5}{V_A(F)} \right) \right) + \frac{1}{2} \left( 0.1 \left( -4 + \overset{0.5x}{V_A(S)} \right) \right) \\ + \frac{1}{2} \left( 0.5 \left( 4 + \overset{0.5}{V_A(F)} \right) \right) + \frac{1}{2} \left( 0.5 \left( 4 + 0.5x V_A(S) \right) \right)$$

Let  $V_{\pi}(s) = V_{\pi}(F) = 0$

$$V_q(S) = \frac{1}{2} (0.8(-8) + 0.2(-8) + 4)$$

$$V_q(F) = \frac{1}{2} (0.9(-4) + 0.1(-4) + 0.5(4) + 0.5(4))$$

$$V_q(S) = -2, \text{ theta} = 0.3$$

$$V_q(F) = 0, \text{ delta}(\Delta_F) = 0$$

$\text{delta}(\Delta_S) = 0$  or some minimum value.  
loop while (1):

loop for  $S \in (-S, F)$ ,  $V_q(S) = V_q(F) = 0$ .

From (A) and (B)

calculate value functions

~~delta~~

$$\Delta_S \leftarrow \max(\Delta_S, |V - V(S)|)$$

Similarly calculate for

$$\Delta_F \leftarrow \max(\Delta_F, |V - V(F)|)$$

condition:

loop until  $\Delta < 0$

So, for next iteration we have

$$V_q(S) = -2, V_q(F) = 0$$

$$V_q(S) = \frac{1}{2} (0.8(-8) + 0.2(-8 + 0.5(-2)) + 4 + 0.5(-2))$$

$$V_q(S) = \frac{1}{2} (0.8(-8) + 0.2(-8 + 0.5(-2)) + 4 + 0.5(-2))$$

$$V_q(F) = \frac{1}{2} (0.9(-4) + 0.1(-4 + 0.5(-2)) + 0.5(4) + 0.5(4 + 0.5(-2)))$$

$$V_q(S) = -2.6$$

$$V_q(F) = -0.3$$

$$\Delta_S = 0.6 \quad \Delta = \max(\Delta_S, \Delta_F)$$

$$\Delta_F = 0.3$$

$$\Delta = 0.6$$

$$V_a(S) = -2.6, V_a(F) = -0.3$$

3. time step 3

$$V_a(s) = \frac{1}{2} \left( 0.5(-1 + 0.5(-0.3)) + \dots \right)$$

$$v_T(s) = \frac{1}{2} \left[ 0.8(-8 + 0.5(-0.3)) + 0.2(-8 + 0.5(-2.6)) + 4 + 0.5(-2.6) \right]$$

$$V_4(F) = \frac{1}{2} \left[ 0.9(-4 + 0.5(-0.3)) + 0.1(-4 + 0.5(-2.6)) \right. \\ \left. + 0.5(4 + 0.5(-0.3)) + 0.5(4 + 0.5(-2.6)) \right]$$

$$v_T(s) = -2.84$$

$$\Delta_S = \max(\Delta_S, V - V_Q(S))$$

$$V_q(F) = -0.5$$

$$\Delta P = \max (A_{F, P}, |V - V_G(P)|)$$

$$\Delta = \Delta$$

$\Delta = 0.24$  [max diff updated value between transition state]

Therefore after 3 iteration, the  $\Delta$  is less than  
taken  $\theta = 0.3$  ~~(max)~~

## Policy improvement step

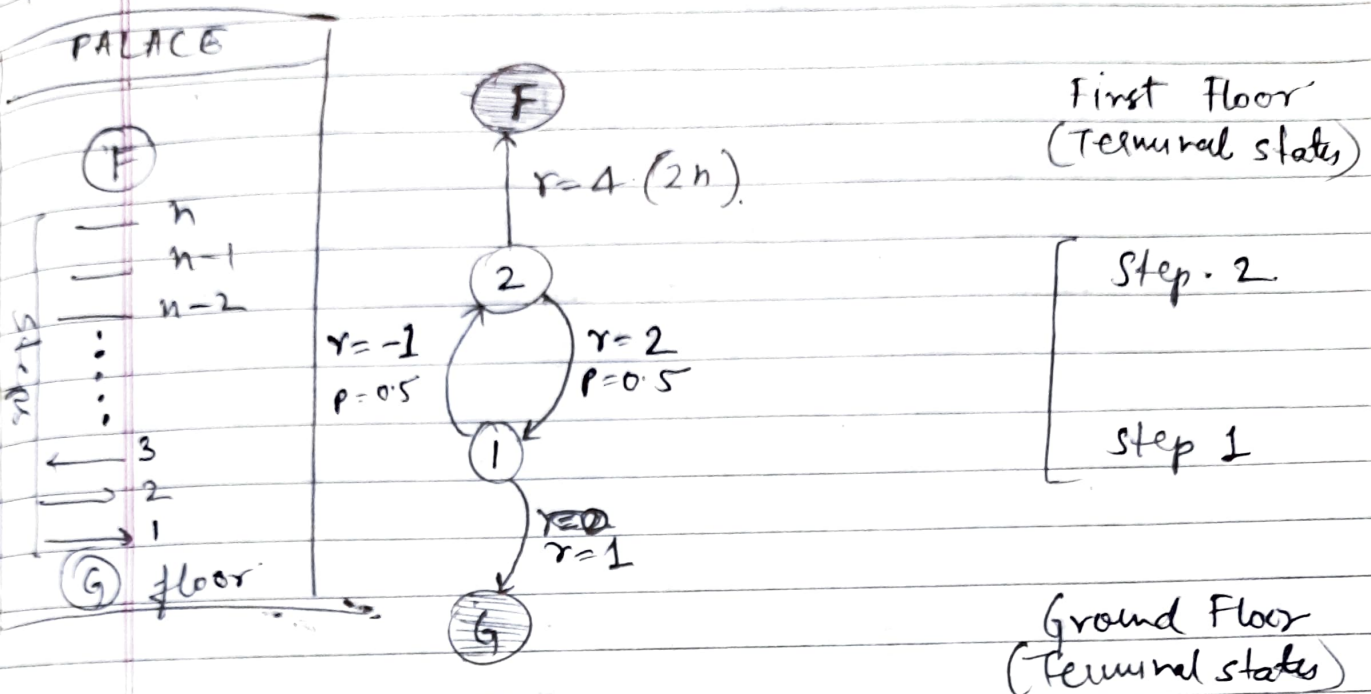
So, to choose policy, (optimal), we pick the  
we action state with max value.

$$\pi^*(s) = \max(V_q(s), V_q(F))$$

$\pi(s) = \forall \pi(F)$  is maximum,  
hence choosing Fresh Information, will have  
optimal policy.



Q15 - MDP for  $n=2$ , Stair well.



**MDP.**

discount factor  $\gamma = 1$ , where  $r$  represents reward.

Following policy  $\pi$  (where  $\pi(\text{up} | s) = 0.5$   
 $\pi(\text{down} | s) = 0.5$ ),

robot can start in any of the step (1 or 2)  
 so, given we have

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

we have

$$V_{\pi}(s=1) = \frac{1}{2} \times (0.5(-1 + V_{\pi}(2)) + 0.5(1 + V_{\pi}(G)))$$

$$V_{\pi}(s=2) = \frac{1}{2} \times (0.5(2 + V_{\pi}(1)) + 0.5(4 + V_{\pi}(F)))$$

considering the values of terminal states as 0.

$$V_{\pi}(G) = V_{\pi}(F) = 0 \quad \text{--- (1)}$$

Solving the equations we have,

$$V_T(1) = 0.25 V_T(2) + V_T(6) + 0.25$$

$$V_A(2) = 1.5 + 0.25 V_A(1) + 0.25 V_A(F)$$

From ①

$$V_q(1) = 0.25 V_q(2)$$

$$V_q(2) = 1.5 + 0.25 V_q(1)$$

Solving (A) and (B) we get -

$$v_d(1) = 1.6$$

$$v_{el}(2) = 1.9$$

Transitions from step ① to step ② and step ② to step ③. give the state values for steps.

optimal policy for actions up and down.  
from step ① and ② respectively.

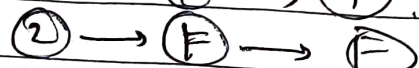
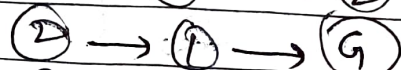
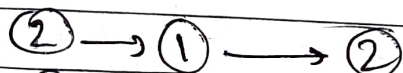
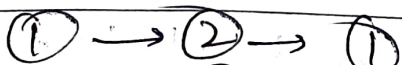
$$a(s) = \arg\max (V_a(i), V_a(2))$$

$$q(s) = vq'(z)$$

So, agent must start from step 2 for single iteration to get max reward return.

Similarly for iterations  $> 1$ , have multiple possibilities.

① starty from state (step ①)



terrible state transitions for two time steps.

$$s_t \rightarrow s_{t+1} \rightarrow s_{t+2}$$

When no. of steps increases  $n > 2$ ,  
 possibility of initial-starting state increases,

As minimum only 1 iteration is needed to  
 reach terminal states for each step  $n=2$ .

No. of iteration needed to reach terminal state  
 increases.

Eg -  $n=3$



Starts for 1 step, closer to G, only 1 iteration,  
 for, F,  $1 \rightarrow 2 \rightarrow 3 \rightarrow F$ , 3 iterations needed,  
 similarly,

No. of iterations increases with  $n$ .

Possible no. of transition with 3 iteration.

From ①,

$1 \rightarrow 2 \rightarrow 1 \rightarrow G$   
 $1 \rightarrow 2 \rightarrow 1 \rightarrow 2$   
 $1 \rightarrow 2 \rightarrow 3 \rightarrow 2$   
 $1 \rightarrow 2 \rightarrow 3 \rightarrow F$

From ②

$2 \rightarrow 1 \rightarrow 2 \rightarrow 1$   
 $2 \rightarrow 1 \rightarrow 2 \rightarrow 3$   
 $2 \rightarrow 3 \rightarrow 2 \rightarrow 1$   
 $2 \rightarrow 3 \rightarrow 2 \rightarrow 3$

Similarly, for step ③, therefore it increases.

• with  $n$  power -

# of iteration  $\approx 2^n + n$  (approx).