MDL ASSIGNMENT - 2.1

MIHIR BANI , 2018911300J

Q1 Transition table

Let S → State (current)
S' → next state
A → Action
P → Probability
T → terminal state (without reward R)

R → Right
U → up
D → down

Table:

S	A	S'	P
A	R	B	0.8
A	R	A	0.2
A	U	C	0.8
A	U	A	0.2
B	L	A	0.8
B	L	B	0.2
B	U	T	0.8
B	U	B	0.2
C	R	T	0.25
C	R	C	0.75
C	D	A	0.8
C	D	C	0.2

Q2. I think the best path can be A → B → T as it is equally probable to reach B or C from A. But reaching T from C is less probable than reaching T from B. ($0.25_{C \to T} < 0.8_{B \to T}$)

Q.3 Reward = 16, $\gamma = 0.2$, $\epsilon = 0.01$

C	0	16	T
A	0	0	B

Initial state

I 1st iteration

→ A: R → $(0.8)(-1 + 0.2 \times 0) + 0.2(-1 + 0.2 \times 0)$
$= -1$

U → $0.8(-1 + 0.2 \times 0) + 0.2(-1 + 0.2 \times 0)$
$= -1$

A = max $(-1, -1)$
$= -1$

→ B: L → $0.8(-1 + 0.2 \times 0) + 0.2(-1 + 0.2 \times 0)$
$= -1$

U → $0.2(-1 + 0.2 \times 0) + 0.8(-4 + 0.2 \times 16)$
$= -0.84$

B = max $(-1, -0.84)$
$= -0.84$

→ C: R → $0.75(-1 + 0.2 \times 0) + 0.25(-3 + 0.2 \times 16)$
$= -0.7$

D → $0.8(-1 + 0.2 \times 0) + 0.2(-1 + 0.2 \times 0)$
$= -1$

C = max $(-0.7, -1)$
$= -0.7$

-0.84	16
-1	-0.7

II 2nd iteration

→ A: U → $0.2(-1 + (0.2)(-1)) + 0.8(-1+(0.2)(-0.7))$.

$= -1.152$

R → $0.2(-1+(0.2)(-1)) + 0.8(-1+0.2)(-0.84))$

$= -1.1744$

$A = \max(-1.152, -1.1744)$

$= -1.152$

→ B: L → $0.8(-1+0.2(-1)) + 0.2(-1+0.2(-0.84))$

$= -1.1936$

U → $0.2(-1+0.2(-0.84)) + 0.8(-4+0.2(16))$

$= -0.8736$

$B = -0.8736$

→ C: R → $0.75(-1+0.2(-0.7)) + 0.25(-3+0.2(16))$

$= -0.805$

D → $0.8(-1+0.2(-1)) + 0.2(-1+0.2(-0.7))$

$= -1.188$

⇒ C = -0.805

-.805	16
-1.152	-0.8736

III 3rd iteration

→ A: U → $0.2(-1+0.2(-1.152)) + 0.8(-1+0.2(-0.805))$

$= -1.17488$

R → $0.2(-1+0.2(-1.152)) + 0.8(-1+0.2(-0.8736))$

$= -1.18586$

$A = \max(-1.17488, -1.18586)$

$= -1.17488$

→ B : L → $0.8(-1 + 0.2(-1.152)) + 0.2(-1 + 0.2(-0.8736))$

$= -1.21926$

U → $0.2(-1 + 0.2(-0.8736)) + 0.8(-4 + 0.2(16))$

$= -0.874944$

$B = \max(-1.21926, -0.874944)$

$B = -0.874944$

→ C : R → $0.75(-1 + 0.2(-0.805)) + 0.25(-3 + 0.2(16))$

$= -0.82075$

D → $0.8(-1 + 0.2(-1.152)) + 0.2(-1 + 0.2(-0.805))$

$= -1.21652$

$C = \max(-0.82075, -1.21652)$

$C = -0.82075$

-0.82	16
-1.1748	-0.875

IV 4th iteration

→ A : U → $0.2(-1 + 0.2(-1.17488)) + 0.8(-1 + 0.2(-0.82075))$

$= -1.17832$

R → $0.2(-1 + 0.2(-1.17488)) + 0.8(-1 + 0.2(-0.874944))$

$= -1.18699$

$A = \max(-1.17832, -1.18699)$

$= -1.17832$

→ B : L → $0.8(-1 + 0.2(-1.17488)) + 0.2(-1 + 0.2(-0.8749))$

$= -1.22298$

U → $0.2(-1 + 0.2(-0.874944)) + 0.8(-4 + 0.2(16))$

$= -0.874998$

$B = \max(-1.22298, -0.874998)$

$B = -0.874998$

C: R → $0.75[-1 + 0.2(-0.82075)) + 0.25(-3 + 0.2(16))$

$= -0.823112$

D → $0.8[-1 + 0.2(-1.17488)) + 0.2[-1 + 0.2(-0.82075))$

$= -1.22081$

D = max $(-0.823112, -1.22081)$

D = -0.823112 .

-0.823112	16
-1.17832	-0.874998

Summary

0	16
0	0

→

-0.84	16
-1	-0.7

→

-0.805	16
-1.152	-0.8736

↓

-0.82	16
-1.1748	-0.875

←

-0.823	16
-1.178	-0.875

Q4. The optimal path ~~after~~ after Value Iteration is

A → C → T

and this is different than what I answered in Q1 (A → B → T)

Q4. The optimal path found after the VI algo is

A → C → T , and this is different from my initial answer of A → B → T.

It was because of the values of step cost and maybe because of the value of discounting factor.

Q.5 The part in the path from A to T is based on the transitions from B → T and C → T. As given the probability of B → T is high (0.8) but the step cost is also high (-4) than the prob. of C → T, (0.25) and the step cost (-3) is lower. Thus we can say that when the reward of final state is below a threshold, the preffered state will be the one that minimizes the step cost (magnitude) so it will try to conserve its total value. But when the final reward is high, then we can risk the high step-cost of B → T, because of greater probability there is now much larger chance of increasing the value. Whereas C → T will have very low probability and the benefit of high Reward will be wasted.

For Reward = 16, optimal path = A → C → T

∃ γ, s.t. γ > 16
and Reward > γ
optimal path will be A → B → T