Dilan Dinushka
School of Computing and Information Systems
Singapore Management University

September 28, 2024

ASSIGNMENT — 1

Question 1

The main assumption need to tackle this problem is, probability of choosing one of the given 3 types of teas are equal. This is required to calculate the overall joint probabilities.

Define the variables as below.

$$\text{Green Tea} = g$$
$$\text{White Tea} = w$$
$$\text{Oolong Tea} = o$$

Let Actual taste tea be denoted by A and Tasted like tea denoted by T. Then the given conditional inputs are as below.

$$P(T = g|A = g) = 0.6 \qquad P(T = w|A = w) = 0.7 \qquad P(T = o|A = o) = 0.8$$
$$P(T = o|A = g) = 0.3 \qquad P(T = o|A = w) = 0.2 \qquad P(T = w|A = o) = 0.1$$
$$P(T = w|A = g) = 0.1 \qquad P(T = g|A = w) = 0.1 \qquad P(T = g|A = o) = 0.1$$

Asking probability is $P(A = o|T = o)$.

$$P(A = o|T = o) = \frac{P(A = o, T = o)}{P(T - o)}$$
$$P(A = o|T = o) = \frac{P(T = o|A = o) * P(A = o)}{P(T = o)}$$

The probability of a given tea taste like Oolong $P(T = o)$ is;

$$\begin{aligned} P(T = o) &= P(T = o|A = o).P(A = o) + P(T = o|A = w).P(A = w) + P(T = o|A = g).P(A = g) \\ &= 0.8 * \frac{1}{3} + 0.2 * \frac{1}{3} + 0.3 * \frac{1}{3} \\ &= 1.3 * \frac{1}{3} \end{aligned}$$

Therefore;

$$P(A = o|T = o) = \frac{P(T = o|A = o) * P(A = o)}{P(T = o)}$$
$$P(A = o|T = o) = \frac{0.8 * \frac{1}{3}}{1.3 * \frac{1}{3}}$$
$$P(A = o|T = o) = 0.615$$

Question 2

The requirement is to find the number of buses needed to use that incurs the least amount of cost.

The given data is as follows;

Number of people = 400
Number of drivers available = 13

Number of large buses (L) = 10
Capacity of a large bus = 40
Cost of a large bus = 750

Number of small buses (S) = 8
Capacity of a small bus = 30
Cost of a small bus = 500

Let number of large buses used be x and number of small buses be y. Then optimization problem is;

$$\begin{array}{ll} \text{minimize} & 750x + 500y \\ \text{subject to} & x \leq 10 \\ & y \leq 8 \\ & x + y \leq 13 \\ & 40x + 30y \geq 400 \\ & x > 0 \\ & y > 0 \end{array}$$

The graph of the given problem is given. The feasible region is surrounded by area covered by the points $(4, 8), (5, 8), (10, 3), (10, 0)$.

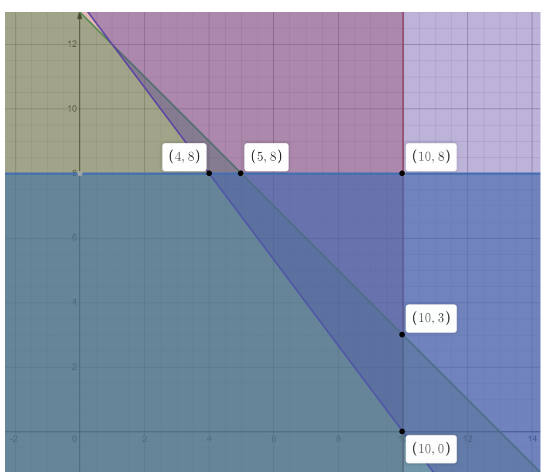


Figure 1: Graph of constraint functions

Here the point $(4, 8)$ yields the optimal lowest cost of 7000. Therefore Having **4 Large buses & 8 Small buses** incurs the lowest optimal cost of 7000.

Question 3

(I)

1. Probability Table for "Age"
This table does not depend on any other variables. Therefore values can be directly calculated.

Age	0	1	2
Prob	0.95	0.32	0.585

2. Probability Table for "Risk Factors"
This table does not depend on any other variables. Therefore values can be directly calculated.

Risk Factors	0	1	2
Prob	0.621	0.265	0.114

3. Probability Table for "Covid-19 Status"
Covid-19 Status value depends on the values from the Age and Risk Factors. Therefore need to build the conditional probability table by considering the each value combination of parent factor variables and averaging across the total values.

	COVID19 Status	0	1	2	3
Age	Risk Factors				
0	0	0.215385	0.553846	0.123077	0.107692
	1	0.047619	0.333333	0.047619	0.571429
	2	0.333333	0.222222	0.222222	0.222222
1	0	0.331658	0.065327	0.381910	0.221106
	1	0.493976	0.313253	0.084337	0.108434
	2	0.289474	0.263158	0.368421	0.078947
2	0	0.207283	0.187675	0.400560	0.204482
	1	0.440994	0.248447	0.086957	0.223602
	2	0.149254	0.208955	0.253731	0.388060

4. Probability Table for "Loss of taste and Smells"
Loss of taste and Smell value depends on the values from the Covid-19 Status. Therefore need to build the conditional probability table by considering the each value of it and averaging across the total values.

Loss of Taste or Smell	0	1
COVID19 Status		
0	0.233677	0.766323
1	0.120930	0.879070
2	0.521277	0.478723
3	0.306604	0.693396

5. Probability Table for "Cough"
Cough value depends on the values from the Covid-19 Status. Therefore need to build the conditional probability table by considering the each value of it and averaging across the total values.

Cough	0	1
COVID19 Status		
0	0.804124	0.195876
1	0.437209	0.562791
2	0.492908	0.507092
3	0.688679	0.311321

6. Probability Table for "Tested Result"

Tested Result value depends on the values from the Covid-19 Status. Therefore need to build the conditional probability table by considering the each value of it and averaging across the total values.

Tested Result	0	1	2
COVID19 Status			
0	0.312715	0.192440	0.494845
1	0.293023	0.427907	0.279070
2	0.095745	0.514184	0.390071
3	0.094340	0.792453	0.113208

(II)

The value for $P(\text{Risk Factors}|\text{Loss of Taste or Smell} = 1, \text{Cough} = 0)$ is:

$$\text{Risk Factor} = 0 \longrightarrow 0.58760363$$
$$\text{Risk Factor} = 1 \longrightarrow 0.30105133$$
$$\text{Risk Factor} = 2 \longrightarrow 0.11134504$$

(III)

The Most Probable Explanation(MPE) for someone who has cough and is 35 years old is that:

$$\text{Tested Result} = 1$$
$$\text{Loss of Taste or Smell} = 0$$
$$\text{COVID19 Status} = 2$$
$$\text{Risk Factors} = 0$$

** Related code files are given.

Question 4

(I)

Let B be the set of all the blocks where $b_i \in B$ denotes any single block i within the set and n denotes the number of blocks. The state space consist of possible locations the set of blocks can be placed.

Let $L = \{b_1, b_2, b_3, ...b_n\} \cup \{t\}$ is all the possible placements of blocks where $b_1, b_2, b_3, ...b_n$ denotes the each of the block's top and t denotes the table top.

Let the state of block i on location j denoted by P_{b_i,l_j}. The all the possible states for the considering blocks can be denoted as below.

State Space $S = \{P_{b_i,l_j}\}$ $\forall b_i \in B$ and $i \neq j$ and $(P_{*,l_i} \neq P_{*,l_j}$ or $l_i = l_j = t)$

Initial State: This can be any state from above state space which satisfy the provided conditions.

- $i \neq j \longrightarrow$ Block cant be placed on itself.
- $P_{*,l_i} \neq P_{*,l_j}$ or $l_i = l_j = t \longrightarrow$ Two blocks cant be on same location or they must be on the table top.

Goal State: Any valid state as defined above.

In the problem an operation can be moving a block from its current location to another valid location.

Therefore an Operation of moving block b_i from location l_j to l_k can be defined as;

Operator: $O(b_i, l_j, l_k) = P_{b_i,l_j} \rightarrow P_{b_i,l_k}$ where $j \neq k$

Here b_i should be either on table-top or should be the top-most block in its respective pile.

Cost for a operations above can be defined as below.

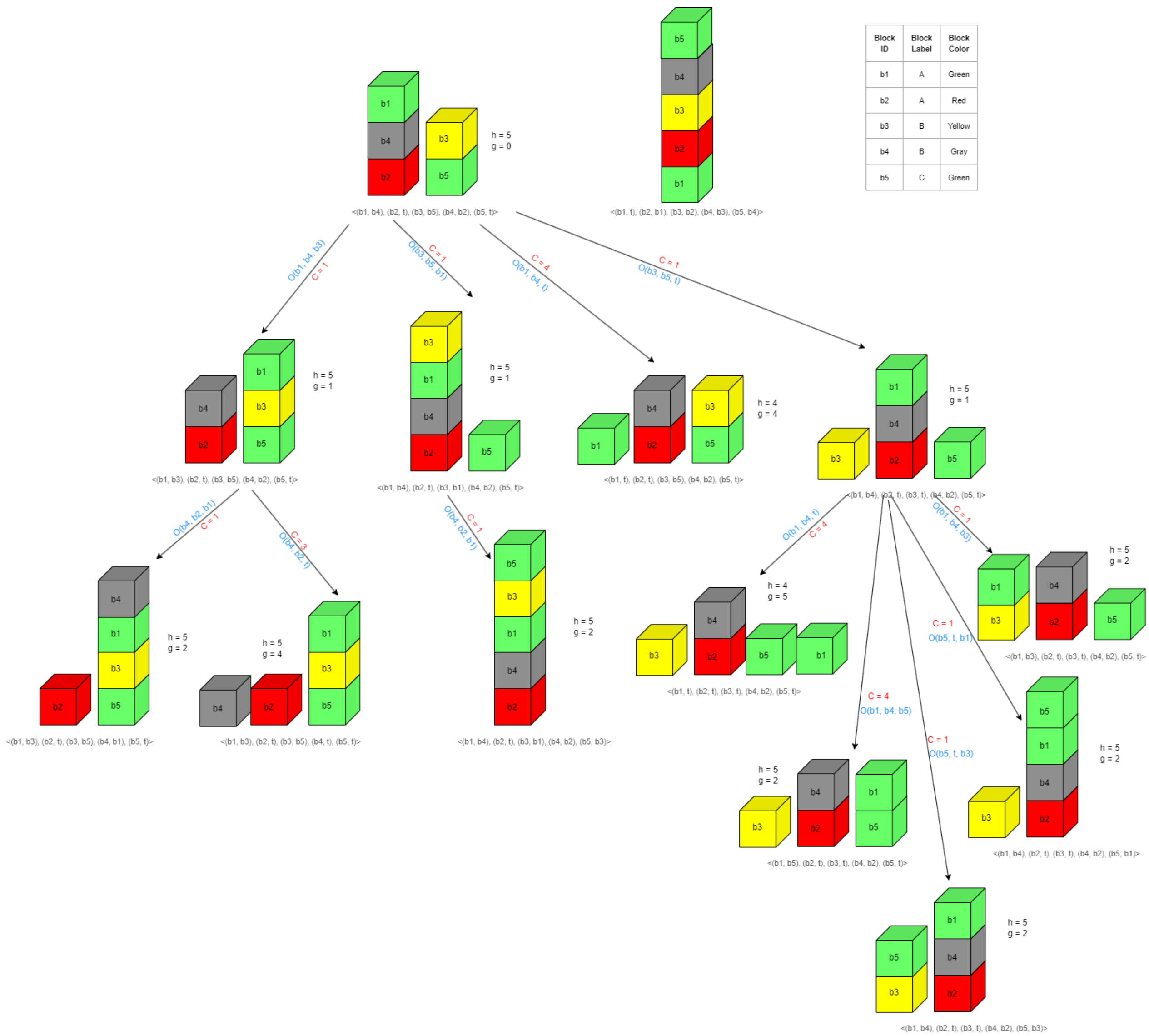
Path Cost: $$C_{O(b_i,l_j,l_k)} = \begin{cases} 1, & \text{if } l_k \neq t \\ 1, & \text{color}(b_i) = \text{Yellow } \& \ l_k = t \\ 2, & \text{color}(b_i) = \text{Orange } \& \ l_k = t \\ 3, & \text{color}(b_i) = \text{Gray } \& \ l_k = t \\ 4, & \text{color}(b_i) = \text{Green } \& \ l_k = t \end{cases}$$

(II)

The block position difference P_{b_i,l_j} between current state and the goal state can be considered as a admissible heuristic in this case because, number of different block positions are always less than or equal to the minimum block positions we are required to change to achieve the goal state.

Heuristic from Current State to Goal State:

$$H(S_c, S_g) = |\{P^c_{b_i,l_j} \forall b_i \in B\} - \{P^g_{b_i,l_j} \forall b_i \in B\}|$$



Block ID	Block Label	Block Color
b1	A	Green
b2	A	Red
b3	B	Yellow
b4	B	Gray
b5	C	Green

Figure 2: Steps of A* search on the given problem

Question 5

The related Markov Field can be illustrated as follows.

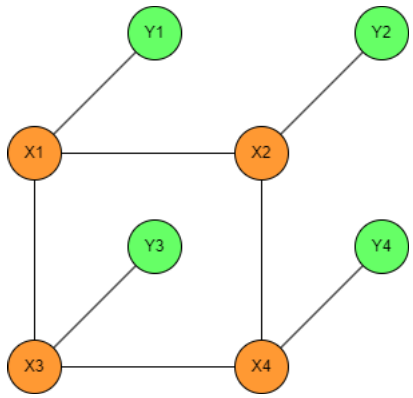


Figure 3: MRF of 4x4 Original Image grid and Observed states

Above field contains 8 2-cliques.

The asked question is $\frac{Pr(X1=W,X2=B|Y1=B,Y2=W,Y3=B,Y4=W)}{Pr(X1=B,X2=W|Y1=B,Y2=W,Y3=B,Y4=W)}$. This get simplified to below form using the Bayes formula.

$$= \frac{Pr(X1=W,X2=B,Y1=B,Y2=W,Y3=B,Y4=W)}{Pr(Y1=B,Y2=W,Y3=B,Y4=W)}$$
$$* \frac{Pr(Y1=B,Y2=W,Y3=B,Y4=W)}{Pr(X1=B,X2=W,Y1=B,Y2=W,Y3=B,Y4=W)}$$
$$= \frac{Pr(X1=W,X2=B,Y1=B,Y2=W,Y3=B,Y4=W)}{Pr(X1=B,X2=W,Y1=B,Y2=W,Y3=B,Y4=W)}$$

We can divide above into two sub problems.

1). $Pr(X1=W,X2=B,Y1=B,Y2=W,Y3=B,Y4=W)$

$$= Pr(X1=W,X2=B,Y1=B,Y2=W,Y3=B,Y4=W)$$

2). $Pr(X1=B,X2=W,Y1=B,Y2=W,Y3=B,Y4=W)$

$$= Pr(X1=B,X2=W,Y1=B,Y2=W,Y3=B,Y4=W)$$

Here Z denotes the normalization term that include sum product probabilities of all possible outcomes. We do not require to calculate this as the question is to find a ratio between two probabilities which would cancel out the Z term.

The potential function values for each of the pixels are as follows.

Xi	Xj	P
W	W	2.0
W	B	0.5
B	W	0.5
B	B	2.0

Xi	Yi	P
W	W	4.0
W	B	0.5
B	W	0.5
B	B	4.0

In this MRF problem, joint probability distribution can be defined using the clique potentials as below.

$$\begin{aligned}&J(X1,X2,X3,X4,Y1,Y2,Y3,Y4)\\&\quad=\frac{1}{Z}*P(X1,X2)P(X2,X3)P(X3,X4)P(X1,X4)P(X1,Y1)P(X2,Y2)P(X3,Y3)P(X4,Y4)\end{aligned}$$

To calculate the answers for sub-problems we need to marginalize the joint probability function with respect to $X3,X4$ values.

$$\begin{aligned}&=\frac{1}{Z}*P(X1,X2)P(X2,X3)P(X3,X4)P(X1,X4)P(X1,Y1)P(X2,Y2)P(X3,Y3)P(X4,Y4)\\&=\frac{1}{Z}\sum_{X3,X4}P(X1,X2)P(X2,X3)P(X3,X4)P(X1,X4)P(X1,Y1)P(X2,Y2)P(X3,Y3)P(X4,Y4)\\&=\frac{1}{Z}P(X1,X2)P(X1,Y1)P(X2,Y2)\sum_{X3,X4}P(X2,X3)P(X3,X4)P(X1,X4)P(X3,Y3)P(X4,Y4)\\&=\frac{1}{Z}P(X1,X2)P(X1,Y1)P(X2,Y2)\sum_{X4}P(X1,X4)P(X4,Y4)\sum_{X3}P(X2,X3)P(X3,X4)P(X3,Y3)\\&\longrightarrow(1)\end{aligned}$$

Calculating the $\sum_{X3}P(X2,X3)P(X3,X4)P(X3,Y3)$ value.

$$\begin{aligned}&=\sum_{X3}P(X2,X3)P(X3,X4)P(X3,Y3)\\&=P_{X3}(X2,W)P_{X3}(W,X4)P_X3W,Y3+P_{X3}(X2,B)P_{X3}(B,X4)P_X3B,Y3\end{aligned}$$

Now calculating the $\sum_{X4}P(X1,X4)P(X4,Y4)\sum_{X3}P(X2,X3)P(X3,X4)P(X3,Y3)$;

$$\begin{aligned}&=\sum_{X4}P(X1,X4)P(X4,Y4)\sum_{X3}P(X2,X3)P(X3,X4)P(X3,Y3)\\&=\sum_{X4}P(X1,X4)P(X4,Y4)\Big(P_{X3}(X2,W)P_{X3}(W,X4)P_X3W,Y3+P_{X3}(X2,B)P_{X3}(B,X4)P_X3B,Y3\Big)\\&=P_{X4}(X1,W)P_{X4}(W,Y4)\Big(P_X3(X2,W)P_{X3,X4}(W,W)P_X3W,Y3+P_{X3}(X2,B)P_{X3,X4}(B,W)P_X3B,Y3\Big)\\&\quad+P_{X4}(X1,B)P_{X4}(B,Y4)\Big(P_X3(X2,W)P_{X3,X4}(W,B)P_X3W,Y3+P_{X3}(X2,B)P_{X3,X4}(B,B)P_X3B,Y3\Big)\\&=P_{X4}(X1,W)P_{X4}(W,Y4)\Big(P_{X3}(X2,W)2P_{X3}(W,Y3)+P_{X3}(X2,B)0.5P_X3B,Y3\Big)\\&\quad+P_{X4}(X1,B)P_{X4}(B,Y4)\Big(P_{X3}(X2,W)0.5P_{X3}(W,Y3)+P_{X3}(X2,B)2P_X3B,Y3\Big)\\&=P_{X4}(X1,W)P_{X4}(W,Y4)\Big(2*P_{X3}(X2,W)P_{X3}(W,Y3)+0.5*P_{X3}(X2,B)P_X3B,Y3\Big)\\&\quad+P_{X4}(X1,B)P_{X4}(B,Y4)\Big(0.5*P_{X3}(X2,W)P_{X3}(W,Y3)+2*P_{X3}(X2,B)P_X3B,Y3\Big)\longleftarrow(E)\end{aligned}$$

Applying above to equation (1);

$$= \frac{1}{Z}P(X1,X2)P(X1,Y1)P(X2,Y2)\sum_{X4}P(X1,X4)P(X4,Y4)\sum_{X3}P(X2,X3)P(X3,X4)P(X3,Y3)$$
$$= \frac{1}{Z}P(X1,X2)P(X1,Y1)P(X2,Y2)*E$$

Where E is the marginalized potentials over $X3, X4$ values.

Therefore, answer to sub problem **(1)**;

$$= \frac{1}{Z}P(X1,X2)P(X1,Y1)P(X2,Y2)*E$$
$$= \frac{1}{Z}P(X1{=}W,X2{=}B)P(X1{=}W,Y1{=}B)P(X2{=}B,Y2{=}W)*\Big(P_{X4}(X1{=}W,W)P_{X4}(W,Y4{=}W)\Big(2 * P_{X3}(X2{=}B,W)P_{X3}(W,Y3{=}B)+0.5*P_{X3}(X2{=}B,B)P_X3B,Y3{=}B\Big)$$
$$+P_{X4}(X1{=}W,B)P_{X4}(B,Y4{=}W)\Big(0.5{*}P_{X3}(X2{=}B,W)P_{X3}(W,Y3{=}B)+2{*}P_{X3}(X2{=}B,B)P_X3B,Y3{=}B\Big)\Big)$$
$$= \frac{1}{Z}0.5*0.5*0.5*\Big(2*4\Big(2*0.5*0.5+0.5*2*4\Big)+0.5*0.5\Big(0.5*0.5*0.5+2*2*4\Big)\Big)$$
$$= \frac{1281}{Z*256}$$

Answer to sub problem **(2)**;

$$= \frac{1}{Z}P(X1,X2)P(X1,Y1)P(X2,Y2)*E$$
$$= \frac{1}{Z}P(X1{=}B,X2{=}W)P(X1{=}B,Y1{=}B)P(X2{=}W,Y2{=}W)*\Big(P_{X4}(X1{=}B,W)P_{X4}(W,Y4{=}W)\Big(2 * P_{X3}(X2{=}W,W)P_{X3}(W,Y3{=}B)+0.5*P_{X3}(X2{=}W,B)P_X3B,Y3{=}B\Big)$$
$$+P_{X4}(X1{=}B,B)P_{X4}(B,Y4{=}W)\Big(0.5{*}P_{X3}(X2{=}W,W)P_{X3}(W,Y3{=}B)+2{*}P_{X3}(X2{=}W,B)P_X3B,Y3{=}B\Big)\Big)$$
$$= \frac{1}{Z}0.5*4*4*\Big(0.5*4*\Big(2*2*0.5+0.5*0.5*4\Big)+2*0.5*\Big(0.5*2*0.5+2*0.5*4\Big)\Big)$$
$$= \frac{84}{Z}$$

Therefore, the **answer to the question** $\frac{Pr(X1{=}W,X2{=}B|Y1{=}B,Y2{=}W,Y3{=}B,Y4{=}W)}{Pr(X1{=}B,X2{=}W|Y1{=}B,Y2{=}W,Y3{=}B,Y4{=}W)}$ **is** $\frac{1281*Z}{Z*256*84} = 0.05957$

Question 6

a)

Zones where the robots could find packages and move = $A1, A2, A3, A4$. Therefore possible states can be defined by their Zone.

State space $S = \{A1, A2, A3, A4\}$ where each Ai denoting the respective zone.

Possible Actions for Robots = {WAIT, DELIVER}.

Action space $A = \{W, D\}$ where "D" means moving between zone to deliver packages and "W" means waiting for package. (Assumed that robots moved only when it has a delivery to complete.)

In the given problem, transition probability matrix values take the form $< s, a, s' >$ which denotes robot moving from state s after taking action a and then going to state s'.

eg: Robot found package in Zone A1 and then deliver package to Zone A2. The related transition probability:

$$P(A2|A1, D) = P(A1, D) * P(A2|A1, D) = 0.25 * 0.3 = 0.075$$

eg: Robot did not find package in Zone A1 and move to A2 probability

$$P(A2|A1, D) = 0$$

Transition Probabilities

State	Action	A1	A2	A3	A4
A1	WAIT	1	0	0	0
A2	WAIT	0	1	0	1
A3	WAIT	0	0	1	0
A3	WAIT	0	0	0	1
A1	DELIVER	0.75	0.075	0.1	0.075
A2	DELIVER	0.15	0.7	0.06	0.09
A3	DELIVER	0.14	0.105	0.65	0.105
A3	DELIVER	0.05	0.025	0.025	0.9

Figure 4: Transition probabilities for the

Reward function

$$R(s, a, s') = \text{Delivery Fee} - \text{Travel Cost}$$

The reward will be zero for any transition that is not defined in terms of moving cost.

b)

1) The related code (Jupyter notebook) is given separately.

2)
Here under each iteration, output line format is **State: Action V-value**

```
Random policy initiated!
A1 : DELIVER 0
A2 : DELIVER 0
A3 : WAIT 0
A4 : WAIT 0

Iteration : 1
A1 : DELIVER 8.295930803665913
A2 : DELIVER 10.316924009025398
A3 : DELIVER 0.0
A4 : DELIVER 0.0

Iteration : 2
A1 : DELIVER 32.45983360215776
A2 : DELIVER 33.60653031538624
A3 : DELIVER 34.85303878502098
A4 : DELIVER 27.743145198148945

Iteration : 3
A1 : DELIVER 32.459834506453774
A2 : DELIVER 33.606531202089435
A3 : DELIVER 34.853039660032586
A4 : DELIVER 27.7431460914493
```

Figure 5: Iteration V values for each state

c)

1) The Linear program to solve the same MDP can be denoted as follows.

$$\begin{aligned} \text{minimize} \quad & \sum_{s} V(s) \\ \text{subject to} \quad & V(s) \geq \sum_{s'} P(s'|s,a) * [R(s,a,s') + \gamma V(s')] \forall a \end{aligned}$$

Here constraints will be generated for all the states and their respective actions. Therefore in the given problem we would have $4 * 2 = 8$ constraints. Defined constraints can be checked in the provided code.

2)

The full code related to both questions are attached separately.

Submitted by Dilan Dinushka on September 28, 2024.

```python
DISCOUNT_FACTOR = 0.95

DELIVER_PROBS = np.array([ ...

WAIT_PROBS = np.identity(4)

TRAVELLING_COST = np.array([ ...

SOURCE_DESC_WITH_PACKAGE_FEE = np.array([ ...

reward = SOURCE_DESC_WITH_PACKAGE_FEE  - TRAVELLING_COST

V = cp.Variable(4)

# Objective is to minimize the sum over V values
objective =cp.Minimize(cp.sum(V))

# Building the Constraints
constraints_wait_action = []
constraints_deliver_action = []
for s in range(4):

    temp_wait = 0
    temp_deliver = 0

    for s_prime in range(4):
        #For wait action reward is zero
        temp_wait += WAIT_PROBS[s, s_prime]*(0.0 + DISCOUNT_FACTOR*V[s_prime])
        temp_deliver += DELIVER_PROBS[s, s_prime]*(reward[s, s_prime] + DISCOUNT_FACTOR*V[s_prime])

    constraints_wait_action.append(V[s]>=temp_wait)
    constraints_deliver_action.append(V[s]>=temp_deliver)

constraints = constraints_wait_action + constraints_deliver_action

prob = cp.Problem(objective, constraints)
result = prob.solve()
V.value
```

```
✓ 0.0s
array([32.45984984, 33.60654623, 34.85305452, 27.74316124])
```

Figure 6: Linear Programming based solution