Assignment 2
Of
Modelling & Simulation Lab (CS1052)

Masters of Technology in Computer Science And Engineering

submitted by
Arghya Bandyopadhyay
RollNo. 20CS4103

submitted to
Dr Nanda Dulal Jana
Assistant Professor
Dept. of CSE

National Institute of Technology, Durgapur

Problem 1

Problem statement.

Let dairy firm has three plants located in a state dairy milk production at each plant is as follows:

Plant 1 – 6 million litres

Plant 2 – 1 million litres

Plant 3 – 10 million litres

Each day the firm must fulfill the needs of its four distribution centers milk requirement at each center in as follows:

Distribution center 1 – 4 million litres

Distribution center 2 – 5 million litres

Distribution center 3 – 3 million litres

Distribution center 4 – 2 million litres

Cost of shipping one million litres of milk from each plant to each distribution center is given in following tables in hundered rupees.

	D_1	D_2	D_3	D_4
P_1	2	3	11	7
P_2	1	0	6	1
P_3	5	8	15	9

Distribution center

Plant

(i) Formulate the mathematical model of the problem

(ii) The dairy farm wishes to determine as to how much should be the shipment from which milk plant to which distribution center which initial solutions using North west corner rule?

Problem formulation

	D_1	D_2	D_3	D_4	Supply (availability)
P_1	2	3	11	7	6
P_2	1	0	6	1	1
P_3	5	8	15	9	10
Demand (req.)	7	5	3	2	17/17

Let x_{ij} = number of units of products to be transported from a production facility i ($i \in 1, 2, 3$) to a distribution center j ($j \in 1, 2, 3, 4$).

The transport problem is stated as an LP model as follows:

Minimize (Total transportation cost) = Z

$$= 2x_{11} + 3x_{12} + 11x_{13} + 7x_{14}$$
$$+ 1x_{21} + 0x_{22} + 6x_{23} + 1x_{24}$$
$$+ 5x_{31} + 8x_{32} + 15x_{33} + 9x_{34}$$

subject to the constraints -

$$\left.\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &= 6 \\ x_{21} + x_{22} + x_{23} + x_{24} &= 1 \\ x_{31} + x_{32} + x_{33} + x_{34} &= 10 \end{aligned}\right] \text{Supply}$$

$$\left.\begin{aligned} x_{11} + x_{21} + x_{31} &= 7 \\ x_{12} + x_{22} + x_{32} &= 5 \\ x_{13} + x_{23} + x_{33} &= 3 \\ x_{14} + x_{24} + x_{34} &= 2 \end{aligned}\right] \text{Demand}$$

and $x_{ij} \geq 0$ for $i \in 1,2,3$ & $j \in 1,2,3,4$

In the above LP model, there are $m \times n = 3 \times 4 = 12$ decision variables, x_{ij} and $m + n = 7$ constraint, where m are number of rows & n are the number of column.

<u>Existence</u> of feasible <u>solution</u>:-

A necessary and sufficient condition for a feasible solution to the transportation problem is,

Total supply = Total demand.

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

$\Rightarrow 6 + 1 + 10 = 7 + 5 + 3 + 2$

$\Rightarrow 17 = 17$

The total supply is equal to total demand, so the problem is balanced transportation problem.

Solution

North west corner method.

	D_1	D_2	D_3	D_4	
P_1	2 (6)	3	11	7	~~6~~ $= a_1$ 0
P_2	1 (1)	0	6	1	~~1~~ $= a_2$ 0
P_3	5	8 (5)	15 (3)	9 (2)	~~10~~ $= a_3$ ~~5~~ ~~2~~ 0
	~~7~~ $= b_1$ ~~1~~ 0	~~5~~ $= b_2$ 0	~~3~~ $= b_3$ 0	~~2~~ $= b_4$ 0	

① comparing a_1 & b_1, since $a_1 < b_1$, allocate $x_{11} = 6$. This exhausts the supply at P_1 & leaves 1 unit as unsatisfied demand at D_1

② Move to cell (P_2, D_1). Compare a_2 & b_1, $a_2 = b_1$, allocate x_{21}

③ Move to cell (P_3, D_2). Since supply at P_3 is equal to the demand at D_2, D_3 and D_4, allocate $x_{32} = 5$, $x_{33} = 3$ & $x_{34} = 2$

It satisfies the feasible solution condition i.e $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$.

and it may be noted that allocated cells are 5 which is less than the required ~~number~~ number $(m+n-1)$ i.e $(3+4-1) = 6$. Thus only one condit

is satisfied, so this solution is degenerate solution. The transportat cost associated with this solution is

$$\text{Total cost} = ((2*6)+(1*1)+(8*5)+(15*3)+(9*2))*100$$

$$=(12+1+40+45+18)*100$$

$$= 116 * 100$$

$$= ₹11600$$

Python Code:

```python
import numpy as np

if __name__ == '__main__':
        total_cost = 0
        no_alloc = 0

        # initiatizes the cost matrix
        cm = np.array([
                [9, 12, 9, 6, 9, 10, 5],
                [7, 3, 7, 7, 5, 5, 6],
                [6, 5, 9, 11, 3, 11, 2],
                [6, 8, 11, 2, 2, 10, 9],
                [4, 4, 6, 2, 4, 2, 0]])
        print("Cost Matrix")

        # prints the cost matrix
        print(cm)

        # calculates the no of rows and columns
        r, c = cm.shape
        print("Rows, Columns: (", r - 1, ",", c - 1, ")")

        # slices the cost matrix to get the sum of the demand
        # and supply vectors
        total_demand = np.sum(cm[r - 1, :])
        total_supply = np.sum(cm[:, c - 1])
        if total_demand == total_supply:
                print("Balanced Transportation Problem.")
        else:
                print("Unbalanced Transportation Problem")
        i = 0
        j = 0

        # This loop does allocation to the cells according to
        # the requirement and possible supply
        while (i < r - 1) and (j < c - 1):
                x = min(cm[r - 1, j], cm[i, c - 1])
                cm[r - 1, j] = cm[r - 1, j] - x
                cm[i, c - 1] = cm[i, c - 1] - x
                total_cost = total_cost + x * cm[i, j]
                no_alloc = no_alloc + 1
                if cm[r - 1, j] < cm[i, c - 1]:
                        j = j + 1
                elif cm[r - 1, j] > cm[i, c - 1]:
                        i = i + 1
                else:
                        i = i + 1
                        j = j + 1
                print("Total Cost: ", total_cost)
                print("No of Allocation: ", no_alloc)
```

```
# checks for the condition m+n−1 = no of allocated cells
if ((r − 1) + (c − 1) − 1) == no_alloc and
                total_demand == total_supply:
        print("Non Degenerate & Feasible Solution")
else:
        print("Degenerate Solution")
```

Output:

```
Cost Matrix
[[ 2  3 11  7  6]
 [ 1  0  6  1  1]
 [ 5  8 15  9 10]
 [ 7  5  3  2  0]]
Rows, Columns: ( 3 , 4 )
Balanced Transportation Problem.
Total Cost:  1200
No of Allocation:  1
Degenerate Solution
Total Cost:  1300
No of Allocation:  2
Degenerate Solution
Total Cost:  5300
No of Allocation:  3
Degenerate Solution
Total Cost:  9800
No of Allocation:  4
Degenerate Solution
Total Cost:  11600
No of Allocation:  5
Degenerate Solution
```

Problem 2

Problem statement

A company has four warehouse & six stores. The warehouse altogether have a surplus 22 units of a given commodity. Individual surplus at warehouse 1, 2, 3 and 4 are 5, 6, 2 and 9 units respectively. The six stores altogether need 22 unit of commodity. Individual requirement stores 1, 2, 3, 4, 5 and 6 are 4, 4, 6, 2, 4 and 2 units respectively. Cost of shipping are unit of commodity from a warehouse i to store j in rupees is given in the matrix below:

		stores					
		1	2	3	4	5	6
warehouse	1	9	12	9	6	9	10
	2	7	3	7	7	5	5
	3	6	5	9	11	3	11
	4	6	8	11	2	2	10

(i) Formulate the mathematical model for the problem.

(ii) Find the IBFS using MWCR.

<u>Problem formulation</u>

	S_1	S_2	S_3	S_4	S_5	S_6	Available
W_1	x_{11} 9	x_{12} 12	x_{13} 9	x_{14} 6	x_{15} 9	x_{16} 10	5
W_2	x_{21} 7	x_{22} 3	x_{23} 7	x_{24} 7	x_{25} 5	x_{26} 5	6
W_3	x_{31} 6	x_{32} 5	x_{33} 9	x_{34} 11	x_{35} 3	x_{36} 11	2
W_4	x_{41} 6	x_{42} 8	x_{43} 11	x_{44} 2	x_{45} 2	x_{46} 10	9
Required	4	4	6	2	4	2	22

Let x_{ij} = number of unit of commodity shipped from warehouse i $(i \in 1, 2, 3, 4)$ to store j $(j \in 1, 2, 3, 4, 5, 6)$

The transport'n problem is stated as LP model as follows:-

Minimizing (Total shipping cost) = Z

$$= 9x_{11} + 12x_{12} + 9x_{13} + 6x_{14} + 9x_{15} + 10x_{16}$$
$$+ 7x_{21} + 3x_{22} + 7x_{23} + 7x_{24} + 5x_{25} + 5x_{26}$$
$$+ 6x_{31} + 5x_{32} + 9x_{33} + 11x_{34} + 3x_{35} + 11x_{36}$$
$$+ 6x_{41} + 8x_{42} + 11x_{43} + 2x_{44} + 2x_{45} + 10x_{46}$$

Subject to the constraint

$$\left.\begin{aligned}
x_{11}+x_{12}+x_{13}+x_{14}+x_{15}+x_{16} &= 5\\
x_{21}+x_{22}+x_{23}+x_{24}+x_{25}+x_{26} &= 6\\
x_{31}+x_{32}+x_{33}+x_{34}+x_{35}+x_{36} &= 2\\
x_{41}+x_{42}+x_{43}+x_{44}+x_{45}+x_{46} &= 9
\end{aligned}\right\}\text{Available}$$

$$\left.\begin{aligned}
x_{11}+x_{21}+x_{31}+x_{41} &= 4\\
x_{12}+x_{22}+x_{32}+x_{42} &= 4\\
x_{13}+x_{23}+x_{33}+x_{43} &= 6\\
x_{14}+x_{24}+x_{34}+x_{44} &= 2\\
x_{15}+x_{25}+x_{35}+x_{45} &= 4\\
x_{16}+x_{26}+x_{36}+x_{46} &= 2
\end{aligned}\right\}\text{Requirement}$$

& for $x_{ij} \geq 0$ for $i=1,2,3,4$ & $j=1,2,3,4,5,6$

for the above LP model, there are $m \times n = 4 \times 6 = 24$ decision variables, x_{ij} and $m+n=10$ constraints, where m are no. of rows and n are the no. of column.

Existance of feasible solution ⇒

Total available = Total requirement

$$\Rightarrow \sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

$$\Rightarrow 5+6+2+9 = 4+4+6+2+4+2$$

$$\Rightarrow 22 = 22$$

The total availability is equal to total requirements, so the problem is balanced transportatⁿ problem.

Solution

North west corner method

	S_1	S_2	S_3	S_4	S_5	S_6	available
W_1	9 (4)	12 (1)	9	6	9	10	$\cancel{5} = a_1$ $\cancel{1}$ 0
W_2	7	3 (3)	7 (3)	7	5	5	$\cancel{6} = a_2$ $\cancel{3}$ 0
W_3	6	5	9 (2)	11	3	11	$\cancel{2} = a_3$ 0
W_4	6	8	11 (1)	2 (2)	2 (4)	10 (2)	$\cancel{9} = a_4$ 8
	$\cancel{4} = b_1$ 0	$\cancel{4} = b_2$ $\cancel{3}$ 0	$\cancel{6} = b_3$ $\cancel{3}$ $\cancel{1}$ 0	$\cancel{2} = b_4$ 0	$\cancel{4} = b_5$ 0	$\cancel{2} = b_6$ 0	

① comparing a_1 & b_1, since $a_1 > b_1$, ; allocate $x_{11} = 4$. This completes the requirement at s_1 and leaves 1 unit as available at w_1

② Move to cell (w_1, s_2) compare a_1 & b_2 i.e $1 < 4$. allocate $x_{12} = 1$ this exhaust the availability at w_1 and leaves 3 unit requirement at s_2.

③ Move to cell (w_2, s_2) compare a_2 & b_2 i.e $6 > 3$, allocate $x_{22} = 3$ this completed the requirement of s_2 & leave 3 unit available at w_2

④ Move to cell (w_2, s_3). compare a_2 & b_3 i.e $3 < 6$, allocate $x_{23} = 3$, this exhausts the availibility of w_2 & leave, 3 unit unsatisfied requirement at s_3.

⑤ Move to cell (w_3, s_3). compare a_3 and b_3 i.e $3 < 6$, allocate

$x_{33} = 2$. This exhausts the availability of w_3 and leave 1 unit unsatisfied requirement at s_2

(6) Move to cell (w_4, s_3), since availability at ~~many~~ w_4 is equal to requirement of s_3, s_4, s_5 and s_6. Therefore allocate $x_{43} = 1$, $x_{44} = 2$, $x_{45} = 4$ and $x_{46} = 2$.

It satisfy the feasible solution conditⁿ i.e $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$, $22 = 22$ and if has allocated cell count 9 which is equal to required $(m+n-1)$ i.e $4+6-1=9$

thus, this satisfy both condition so it is non-degenerate feasible solⁿ.

The shipping cost associated with this solutⁿ is

$$\text{Total cost} = (9\times4)+(12\times1)+(3\times3)+(7\times3)+(9\times2) + (11\times1)+(2\times2)+(4\times2)+(10\times2)$$

$$= 36+12+9+21+18+11+4+8+20$$

$$= ₹\,139$$

Python Code:

```
import numpy as np

if __name__ == '__main__':
        total_cost = 0
        no_alloc = 0

        # initiatizes the cost matrix
        cm = np.array([
        [2, 3, 11, 7, 6],
        [1, 0, 6, 1, 1],
        [5, 8, 15, 9, 10],
        [7, 5, 3, 2, 0]])
        print("Cost Matrix")

        # prints the cost matrix
        print(cm)

        # calculates the no of rows and columns
        r, c = cm.shape
        print("Rows, Columns: (", r - 1, ",", c - 1, ")")

        # slices the cost matrix to get the sum of the demand
        # and supply vectors
        total_demand = np.sum(cm[r - 1, :])
        total_supply = np.sum(cm[:, c - 1])
        if total_demand == total_supply:
                print("Balanced Transportation Problem.")
        else:
                print("Unbalanced Transportation Problem")
        i = 0
        j = 0

        # This loop does allocation to the cells according to
        # the requirement and possible supply
        while (i < r - 1) and (j < c - 1):
                x = min(cm[r - 1, j], cm[i, c - 1])
                cm[r - 1, j] = cm[r - 1, j] - x
                cm[i, c - 1] = cm[i, c - 1] - x
                total_cost = total_cost + x * cm[i, j]
                no_alloc = no_alloc + 1
                if cm[r - 1, j] < cm[i, c - 1]:
                        j = j + 1
                elif cm[r - 1, j] > cm[i, c - 1]:
                        i = i + 1
                else:
                        i = i + 1
                        j = j + 1
                print("Total Cost: ", total_cost * 100)
                print("No of Allocation: ", no_alloc)
```

```python
        # checks for the condition m+n-1 = no of allocated cells
        if ((r - 1) + (c - 1) - 1) == no_alloc and
                        total_demand == total_supply:
                print("Non Degenerate & Feasible Solution")
        else:
                print("Degenerate Solution")
```

Output:

```
Cost Matrix
[[ 9 12  9  6  9 10  5]
 [ 7  3  7  7  5  5  6]
 [ 6  5  9 11  3 11  2]
 [ 6  8 11  2  2 10  9]
 [ 4  4  6  2  4  2  0]]
Rows, Columns: ( 4 , 6 )
Balanced Transportation Problem.

Total Cost:  36
No of Allocation:  1
Degenerate Solution
Total Cost:  48
No of Allocation:  2
Degenerate Solution
Total Cost:  57
No of Allocation:  3
Degenerate Solution
Total Cost:  78
No of Allocation:  4
Degenerate Solution
Total Cost:  96
No of Allocation:  5
Degenerate Solution
Total Cost:  107
No of Allocation:  6
Degenerate Solution
Total Cost:  111
No of Allocation:  7
Degenerate Solution
Total Cost:  119
No of Allocation:  8
Degenerate Solution
Total Cost:  139
No of Allocation:  9
Non Degenerate & Feasible Solution
```