Research Article

Journal of Electrical Electronics Engineering

Solving Multi Objective Transportation Problems using Harmonic Mean and Graph Theory

[1]E.M.D.B. Ekanayake, [2]E.M.U.S.B. Ekanayake

[1]Department of Physical Sciences, Faculty of Applied Sciences

[2]Rajarata University of Sri Lanka, Mihinthale, Sri Lanka

[*]Corresponding author

E.M.U.S.B. Ekanayake, Rajarata University of Sri Lanka, Mihinthale, Sri Lanka

Submitted: 21 Sep 2022; **Accepted:** 17 Oct 2022; **Published:** 24 Nov 2022

Citation: *E.M.D.B. Ekanayake, E.M.U.S.B. Ekanayake*. (2022). Solving Multi Objective Transportation Problems using Harmonic Mean and Graph Theory. J Electrical Electron Eng, 1(1), 115-130.*

Abstract

Operations research is frequently applied to economics activities, where more attention has also been directed to transportation problems (TP). Another application of this type of transportation problem can be shown as multi objective transportation problems (MOTPs). Usually, in a transportation problem, only one objective of transporting goods from source to destination is considered. However, in a MOTP, different objectives such as time, cost, and risk must be considered. The basic requirement of a multi objective transportation problem in such way as to obtain a minimum value for the objectives mentioned above. Therefore, researchers are developing various algorithms to find an efficient solution to these specific problems. Also, in this paper, we focus on presenting a new approach for an efficient solution to the MOTP. It is recommended to build the method based on the harmonic mean algorithm and spanning tree, compare its solutions with the methods of other concurrent researchers and check the effectiveness of this article. We can say that this new approach is an efficient and easy-to-understand method that provides relatively accurate solutions.

Keywords: Efficient Solutions, Harmonic Mean, Multi Objective Transportation Problems, Spanning Tree.

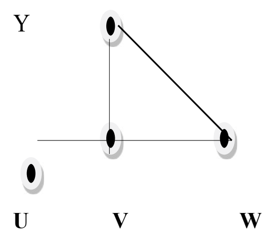
Introduction

Decision-making is a concept that is primarily considered in operational research. There, successful organizational development can be achieved by choosing options that are high in quality and quantity. Here, in order to reach an optimal conclusion, the relevant decision should be based on several specific criteria. Thus, transportation problems can be shown as a mathematical model that facilitates decision-making. The transportation problem model is a widely used case in the field of operations research. In such a case, the primary consideration is to minimize the cost of transfer of goods from suppliers to destination. Thus, presented the first algorithm in 1941 to find the basic feasible solution to the transportation problem. Recently, many researchers have presented and continue to present their research work on transportation problems [1, 2]. However, in real life, the above procedure results in a very different situation. That is, the cost of transportation is only one factor that affects the company's profit. They consider how to minimize many different objectives in transportation, such as time, risk, and energy consumption. In such a case, the multi-objective transportation problem can be introduced as an answer to that problem [3]. While searching for solutions to such problems, multi-objective transportation problems came up. Therefore, researchers developed various methods to come up with a basic, viable solution to the problem [4]. In 1973, using goal programing and 1979 introduced new algorithm for solve the multi objective transportation problem. Also, presented new methods for solving multi-objective transportation problems. In recent times, various researchers have conducted their research work under this category. They can be pointed out as in this paper; we work towards an efficient solution to multi-objective transportation problems. It is primarily based on a spanning tree algorithm. In addition, we look forward to corroborating this result with other research work [5, 10].

Preliminaries

We use the spanning tree concept to solve these transportation problems, by using the depth first search of the construction-spanning tree. Before the discuss about this method we need some definitions studied.so in this section, discussed about some important definitions.

Graph



Definitions

A simple graph G consists of a non-empty finite set V(G) of elements called vertices (or nodes), and a finite set E(G) of distinct unordered pairs of distinct elements of V(G) called edges. We call V(G) the vertex set and E(G) the edge set of G.

Or

A graph consists vertices and edge. Each vertex is represented by an edge join vertices are often called the end point of the edge.

V (G) = {u, v, w, y}
E (G) = {uv, vw, uw, wy}

Bipartite Graph

Two subsets of the vertex set are divided. They are not similar in any way; hence, no two-graph vertices in the same set are adjacent.

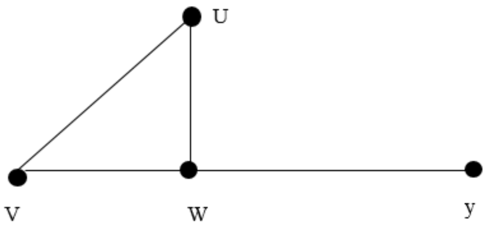
Tree A tree is a connected undirected graph with no cycle.
Spanning Tree A subgraph of graph G is its spanning tree which, to be a tree, has all the vertices of G.

Depth – First Search Method

A spanning tree can be constructed using a depth-first search of the graph by following the steps below.
1) As the root, choose a random vertex from the graph.
2) Create a path from the root by sequentially adding vertices and edges.
3) Continue to add vertices and edges to this path as much as possible.
4) If all vertices are included in the tree, the tree is spanning.

Harmonic Mean

The reciprocal of the average of the reciprocals of the data values is known as the Harmonic Mean (HM). In order to properly balance the data, the harmonic mean assigns smaller weights to large values and larger weights to small ones.



Define the harmonic mean as n set of variants $x_1, x_2, \ldots, x_n$;

$$HM = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}$$

Methodology

Usually, a multi objective transportation problem can be solved easily by giving it a mathematical

To destination→ ↓From source	D_1	D_2	…	D_n	Supply a_i
S_1	C^1_{11} C^2_{11} X_{11} C^r_{11}	C^1_{12} C^2_{12} X_{12} C^r_{12}	…	C^1_{1n} C^2_{1n} X_{1n} C^r_{1n}	a_1
S_2	C^1_{21} X_{21} C^2_{21} C^r_{21}	C^1_{22} C^2_{22} X_{22} C^r_{22}	…	C^1_{2n} C^2_{2n} X_{2n} C^r_{2n}	a_2
⋮	⋮	⋮	⋮	⋮	⋮
S_m	C^1m1 C^2m1 $Xm1$ C^rm1	C^1_{m2} C^2_{m2} X_{m2} C^r_{m2}	…	C^1_{mn} C^2_{mn} X_{mn} C^r_{mn}	a_m
Demand b_j	b_1	b_2	…	b_n	$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$

Table 01: General representation of transportation tableau

Where; X_{ij} - the number of commodities moved from source i to destination j.

C^r_{ij} - the unit cost associated with r^{th} objective moving from source i to destination j.

r -objective of the problem. Like a cost, energy, time etc. (r =1, 2, 3…, r - integer)

Let us consider the transportation problem, number of "m" source to number of "n" destination.in this procedure we denote by transports commodities "$a_1, a_2 ... a_n$" are available supplies in i^{th} source and "$b_1, b_2 ... b_m$' are demand of product in the j^{th} destination.

A mathematical form of a transportation problem involving multiple objective functions is given above. The mathematical model for solving that problem can be described as follows:

Objective Function

$$Min\ Z_r = \sum_{i=1}^{m}\sum_{j=1}^{n} c^r_{ij}.x_{ij}$$

Subjective

$$\sum_{i=1}^{m} X_{ij} = a_i, \quad a_i \geq 0, \qquad (i = 1,2, \dots m)$$

$$\sum_{j=1}^{n} X_{ij} = b_j, \quad b_j \geq 0, \qquad (j = 1,2, \dots n) \qquad X_{ij} \geq 0\ for\ all\ i and\ j$$

Also implies that total supply and total demand must be equal to solve the problem

Proposed Algorithm

Step 1: Convert the given multi-objective transportation problem to a general transportation model using the modified harmonic mean formula. Here, r^{th} the objective values are reduced to one value. (This value is called the weight (w_{ij}) value of the problem.)

$$W_{ij} = \frac{r}{\frac{1}{C^1_{mn}} + \frac{1}{C^2_{mn}} + \dots + \frac{1}{C^r_{mn}}}$$

Step 2: If the problem is unbalanced, add a dummy column or row appropriate and balance it.

Step 3: Formulate the problem in the form of a bipartite graph, taking the values "S_1 to S_m and D_1 to D_n" as vertices and the corresponding W_{ij} value as an edge.

Step 4: - From the bipartite graph, construct the spanning tree using depth-first search.
Construct spanning tree. (Modified step)

Calculate the total weight value of each vertex. If the number of rows and columns is not equal, then take the average weight value. Then use the maximum total weight or maximum average weight value vertex as a starting vertex.

Then Select the next vertex along the edge with the least weight from the selected vertex. Create a spanning tree that includes all vertices in that way.

Step 5: - Then make allocations to the corresponding cells in the flow table according to the order in which the edges of this tree are traversed. If it will not satisfy all the supply and demand of the problem, finish the problem by placing an allocation according.

Step 6: - After that, calculate the respective effective cost values are each objective.

In addition, we can represent details by flow chart below, for proposed new algorithm

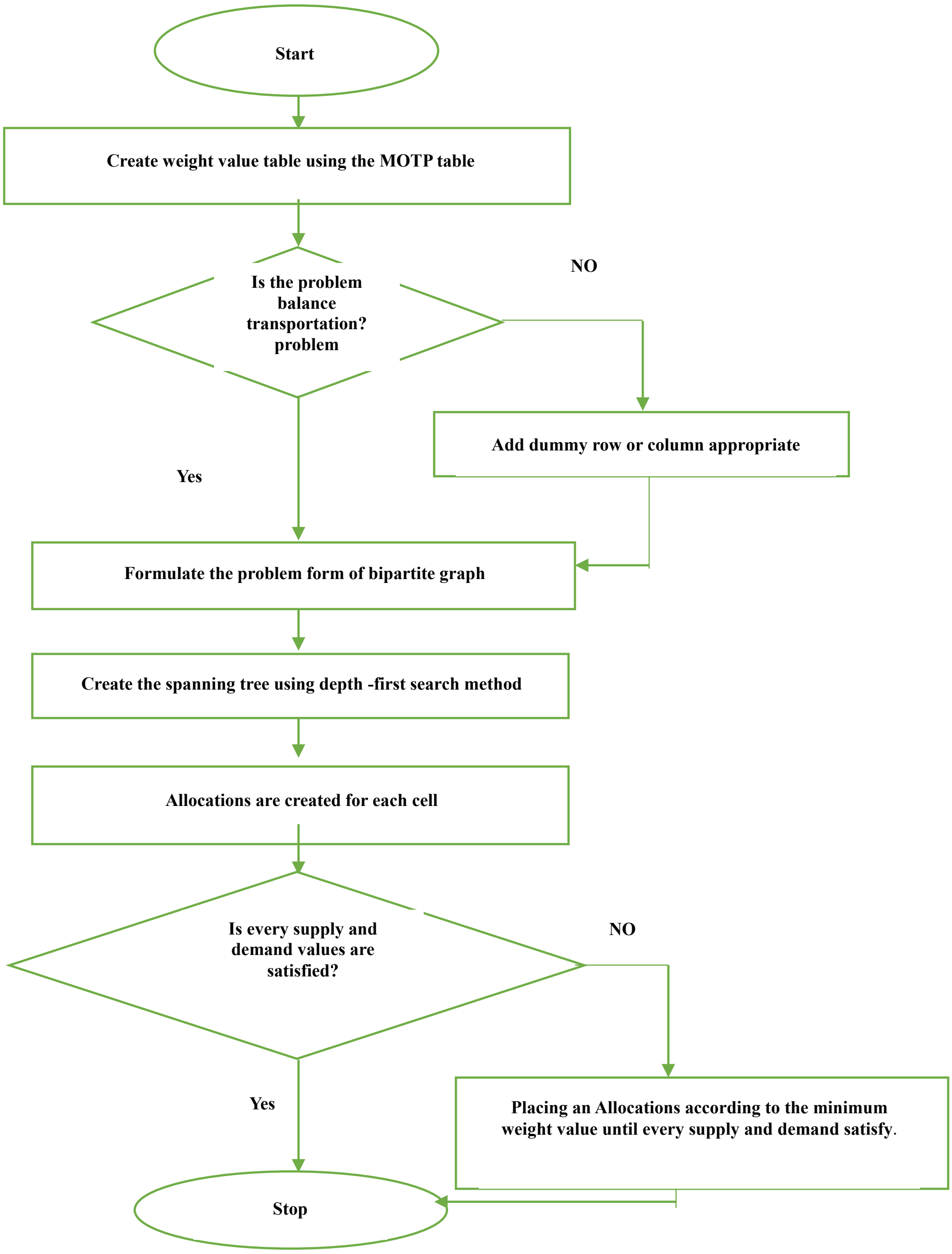


Figure 1: Flowchart of the algorithm

Numerical examples

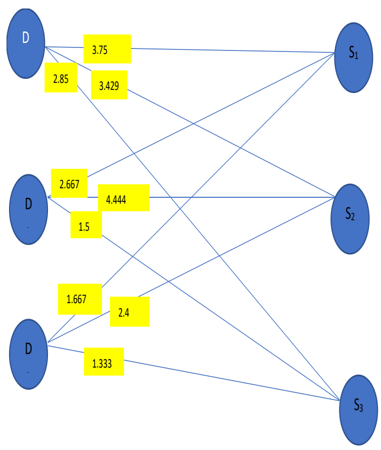
Ex.1

	D_1	D_2	D_3	Supply
S_1	35	42	51	8
S_2	43	54	23	5
S_3	52	13	21	2
Damand	7	4	4	

Step 1)

	D_1	D_2	D_3	Supply
S_1	3.75	2.667	1.667	8
S_2	3.429	4.444	2.4	5
S_3	2.857	1.5	1.333	2
Damand	7	4	4	

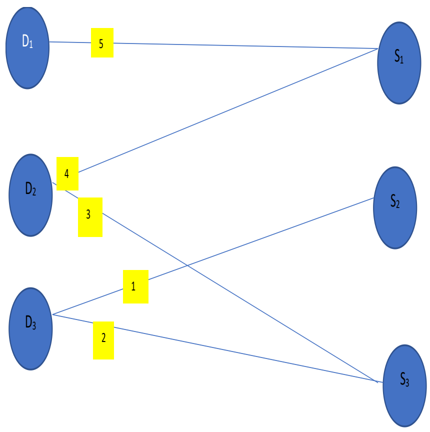
Step 3)



Step 4)

$\sum D_1$	10.036	$\sum S_1$	8.084
$\sum D_2$	8.611	$\sum S_2$	10.273
$\sum D_3$	5.4	$\sum S_3$	5.69

Maximum total weight of edge in the vertex S_2. We construct spanning tree using the Depth first search. Starting vertex S_2 then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step 5 and step 6)

	D_1	D_2	D_3	Supply
S_1	3 6 5	4 2 2	51	8 (6) (0)
S_2	4 1 3	54	2 4 3	5 (1)
S_3	5 2	1 2 1	21	2 (0)
Damand	7 (1) (0)	4 (2) (0)	4 (0)	

$Z_1=(3\times6)+(4\times1)+(4\times2)+(1\times2)+(4\times2)=40$

$Z_2=(5\times6)+(3\times1)+(2\times2)+(3\times2)+(3\times4)=55$

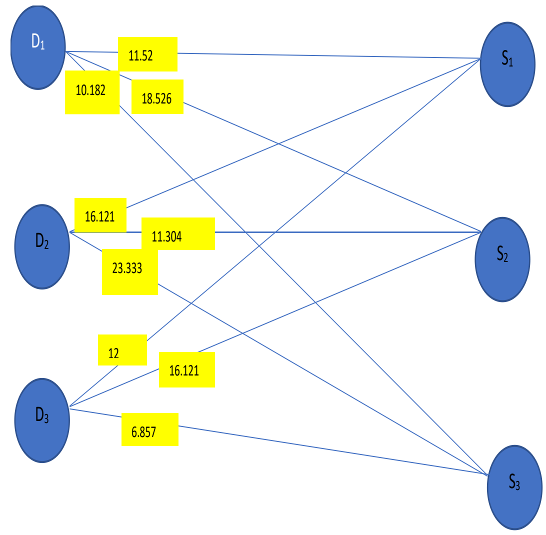
Ex.2

	D_1	D_2	D_3	Supply
S_1	169	1914	1212	14
S_2	2216	1310	1914	16
S_3	148	2820	86	12
Damand	10	15	17	

Step 1)

	D_1	D_2	D_3	Supply
S_1	11.52	16.121	12	14
S_2	18.526	11.304	16.121	16
S_3	10.182	23.333	6.857	12
Damand	10	15	17	

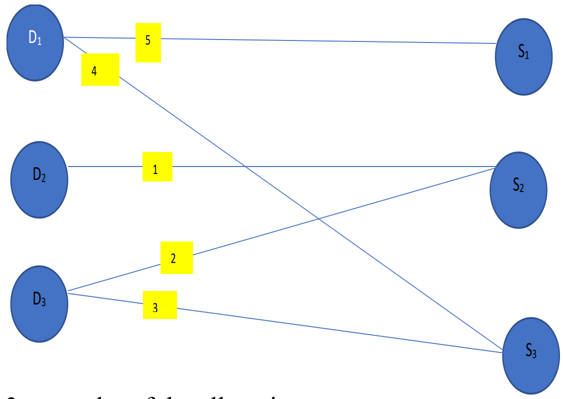
Step 3)



Step 4)

$\sum D_1$	40.278	$\sum S_1$	39.641
$\sum D_2$	50.758	$\sum S_2$	45.951
$\sum D_3$	34.978	$\sum S_3$	40.372

Maximum total weight of edge in the vertex D_2. We construct spanning tree using the Depth first search. Starting vertex D_2 then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step 5 and Step 6)

	D_1	D_2	D_3	Supply
S_1	16 10 9	19 14	12 4 12	14 (4) (0)
S_2	22 16	13 15 10	19 1 14	16 (1) (0)
S_3	14 8	28 20	8 12 6	12 (0)
Damand	10 (0)	15 (0)	17 (16) (4) (0)	

$Z_1 = (16\times10)+(13\times15)+(4\times12)+(1\times19)+(8\times12)=518$

$Z_2 = (9\times10)+(10\times15)+(12\times4)+(14\times1)+(6\times12)=374$

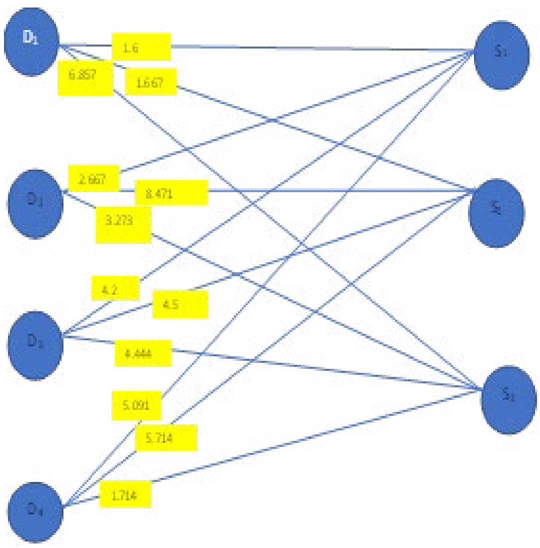
Ex.3

	D_1	D_2	D_3	D_4	Supply
S_1	14	24	73	74	8
S_2	15	98	39	410	19
S_3	86	92	45	61	17
Damand	11	3	14	16	

Step 1)

	D_1	D_2	D_3	D_4	Supply
S_1	1.6	2.667	4.2	5.091	8
S_2	1.667	8.471	4.5	5.714	19
S_3	6.857	3.273	4.444	1.714	17
Damand	11	3	14	16	

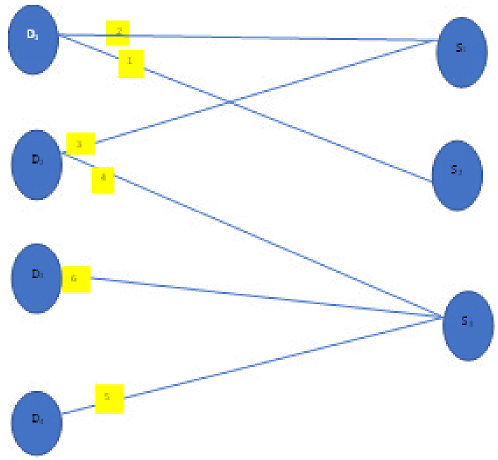
Step 3)



Step 4)

$\sum D_1$	10.124	$\sum S_1$	13.558
$\sum D_2$	14.411	$\sum S_2$	20.352
$\sum D_3$	13.144	$\sum S_3$	16.288
$\sum D_4$	12.519		

Maximum Average weight of edge in the vertex S_2 (5.088). We construct spanning tree using the Depth first search. Starting vertex S_2 and then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step 5 and Step 6)

	D_1	D_2	D_3	D_4	Supply
S_1	1 4	23 4	75	7 4	8 (5) (0)
S_2	1 11 5	9 8	38 9	4 10	19 (8) (0)
S_3	8 6	9 2	4 1 5	6 16 1	17 (1) (0)
Damand	11 (0)	3 (0)	14 (13 (8) (0)	16 (0)	

$Z_1=(2\times3)+(7\times5)+(1\times11)+(3\times8)+(1\times4)+(6\times16)=176$
$Z_2=(2\times4)+(3\times5)+(5\times11)+(9\times8)+(1\times5)+(1\times16)=175$

Ex.4

	D_1	D_2	D_3	D_4	Supply
S_1	24 14	29 21	18 18	23 13	21
S_2	33 24	20 13	29 21	32 23	24
S_3	21 12	42 30	12 9	20 11	18
S_4	8 6	9 2	4 1 5	6 16 1	17 (1) (0)
Damand	11 (0)	3 (0)	14 (13) (8) (0)	16 (0)	

$Z_1=(2\times3)+(7\times5)+(1\times11)+(3\times8)+(1\times4)+(6\times16)=176$
$Z_2=(2\times4)+(3\times5)+(5\times11)+(9\times8)+(1\times5)+(1\times16)=175$

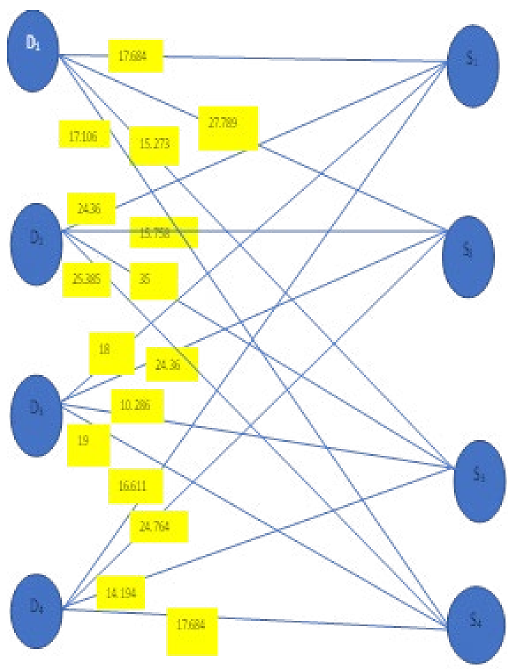
Ex.4

	D₁	D₂	D₃	D₄	Supply
S_1	24 14	29 21	18 18	23 13	21
S_2	33 24	20 13	29 21	32 23	24
S_3	21 12	42 30	12 9	20 11	18
S_4	25 13	30 22	19 19	24 14	30
Damand	15	22	26	30	

Step 1)

	D_1	D_2	D_3	D_4	Supply
S_1	17.684	24.36	18	16.611	21
S_2	27.789	15.758	24.36	26.764	24
S_3	15.273	35	10.286	14.194	18
S_4	17.106	25.385	19	17.684	30
Damand	15	22	26	30	

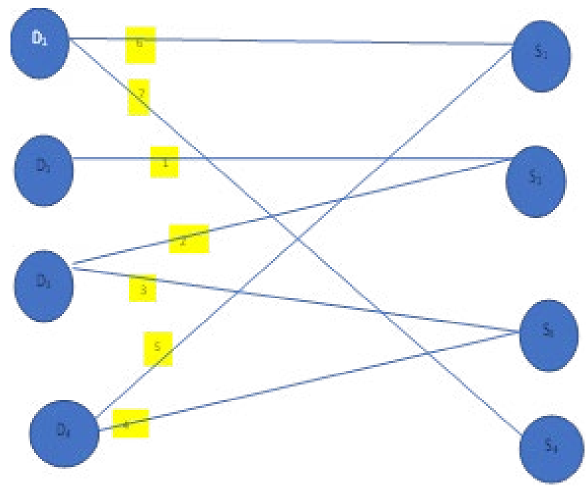
Step 3)



Step 4)

$\sum D_1$	77.852	$\sum S_1$	76.655
$\sum D_2$	100.503	$\sum S_2$	94.671
$\sum D_3$	71.646	$\sum S_3$	74.753
$\sum D_4$	75.253	$\sum S_4$	79.175

Maximum total weight of edge in the vertex D_2. We construct spanning tree using the Depth first search. Starting vertex D_2 and then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step 5 and Step)

	D₁	D₂	D₃	D₄	Supply
S_1	24 14	29 21	18 18	23 21 13	21 (0)
S_2	33 24	20 22 13	29 2 21	32 23	24 (2) (0)
S_3	21 12	42 30	12 18 9	20 11	18 (0)
S_4	25 15 13	30 22	19 6 19	24 9 14	30 (15) (6) (0)
Damand	15 (0)	22 (0)	26 (24) (6) (0)	30 (9) (0)	

$Z_1=(25\times15)+(20\times22)+(29\times2)+(12\times18)+(6\times19)+(23\times21)+(24\times9)=1902$

$Z_2=(13\times15)+(13\times22)+(2\times21)+(9\times18)+(19\times6)+(13\times21)+(9\times14)=1198$

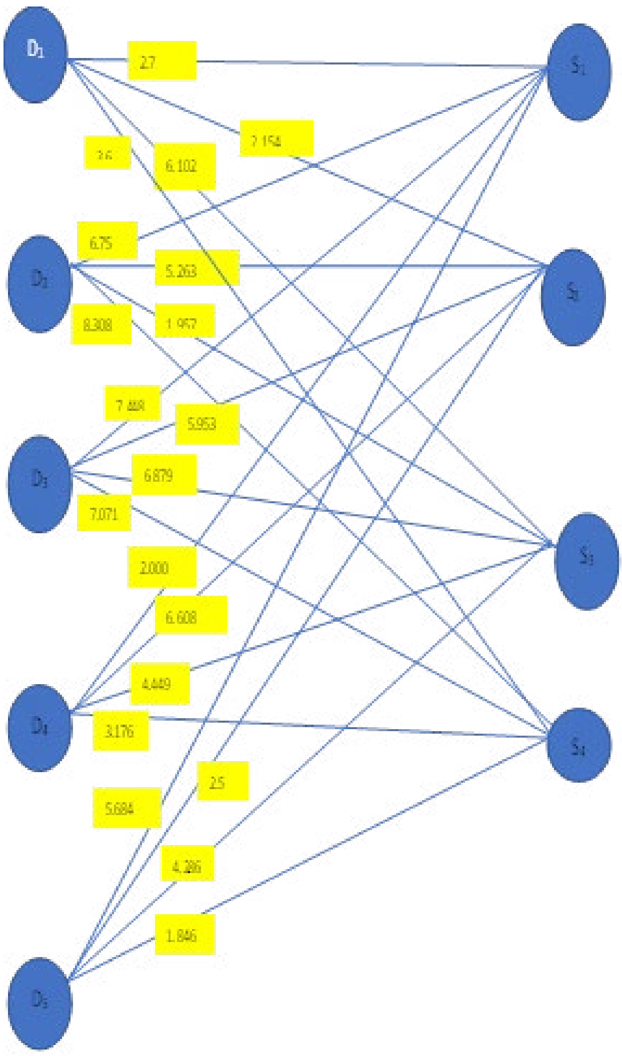
Ex.5

	D₁	D₂	D₃	D₄	D_5	Supply
S_1	922	1294	986	613	946	5
S_2	714	398	794	759	522	4
S_3	685	513	985	1143	356	2
S_4	626	889	1166	293	281	9
Damand	4	4	6	2	4	

Step 1)

	D_1	D_2	D_3	D_4	D_5	Supply
S_1	2.7	6.75	7.448	2	5.684	5
S_2	2.154	5.263	5.953	6.608	2.5	4
S_3	6.102	1.957	6.879	4.449	4.286	2
S_4	3.6	8.308	7.071	3.176	1.846	9
Damand	4	4	6	2	4	

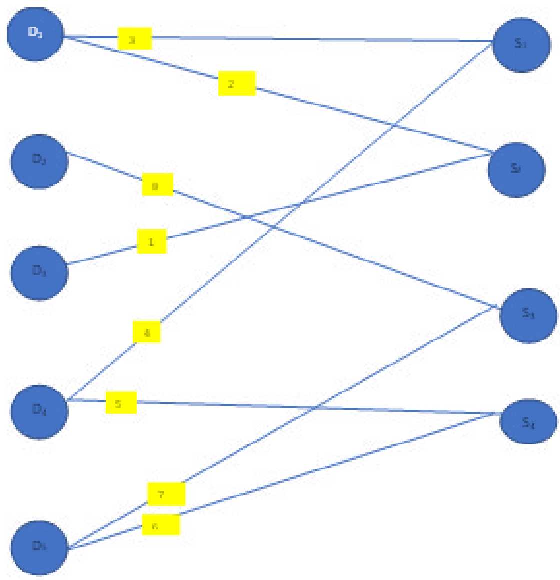
Step 3)



Setp4

$\sum D_1$	14.556	$\sum S_1$	24.582
$\sum D_2$	22.278	$\sum S_2$	22.478
$\sum D_3$	27.351	$\sum S_3$	23.673
$\sum D_4$	16.233	$\sum S_4$	24.001
$\sum D_5$	14.316		

Maximum Average weight of edge in the vertex D_3 (6.838). We construct spanning tree using the Depth first search. Starting vertex D_3 and then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step5 and Step 6)

	D_1	D_2	D_3	D_4	D_5	Supply
S_1	9 4 2 2	12 9 4	9 8 6	6 1 1 3	9 4 6	5 (1) (0
S_2	7 1 4	3 9 8	7 4 9 4	7 5 9	5 2 2	4 (0)
S_3	6 8 5	5 2 1 3	9 8 5	11 4 3	3 5 6	2 (0)
S_4	6 2 6	8 2 8 9	11 2 6 6	2 1 9 3	2 4 8 1	9 (8) (4) (2) (0)
Damand	4 (0)	4 (2) (0)	6 (2) (0)	2 (1) (0)	4 (0)	

$Z_1=(9\times4)+(5\times2)+(8\times2)+(7\times4)+(11\times2)+(2\times1)+(6\times1)+(2\times4)=128$

$Z_2=(2\times4)+(1\times2)+(2\times8)+(9\times4)+(2\times6)+(1\times1)+(9\times1)+(8\times4)=116$

$Z_3=(2\times4)+(3\times2)+(2\times9)+(4\times4)+(2\times6)+(3\times1)+(3\times1)+(1\times4)=70$

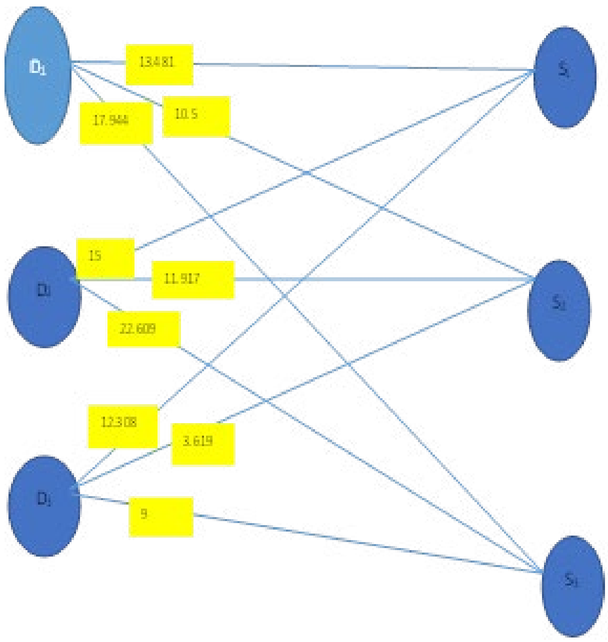
Ex.6

	D_1	D_2	D_3	Supply
S_1	13 14	15 15	16 10	17
S_2	7 21	11 13	2 19	12
S_3	19 17	20 26	9 9	16
Damand	14	8	23	

Step 1)

	D_1	D_2	D_3	Supply
S_1	13.481	15	12.308	17
S_2	10.5	11.917	3.619	12
S_3	17.944	22.609	9	16
Damand	14	8	23	

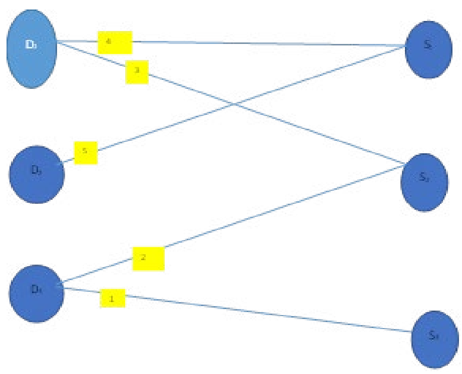
Step 3)



Step 4)

$\sum D_1$	41.925	$\sum S_1$	40.789
$\sum D_2$	49.526	$\sum S_2$	26.036
$\sum D_3$	24.927	$\sum S_3$	49.553

Maximum total weight of edge in the vertex S_3. We construct spanning tree using the Depth first search. Starting vertex S_2 then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step5 and Step 6)

	D_1	D_2	D_3	Supply
S_1	13 9 14	15 8 15	12.308	17 (8) (0)
S_2	7 5 21	11.917	2 7 19	12 (5) (0)
S_3	19 17	20 26	9 16 9	16 (0)
Damand	14 (9) (0)	8 (0)	23 (7) (0)	

$Z_1=(13\times9)+(7\times5)+(15\times8)+(2\times7)+(9\times16)=430$
$Z_2=(14\times9)+(21\times5)+(15\times8)+(19\times7)+(9\times16)=628$

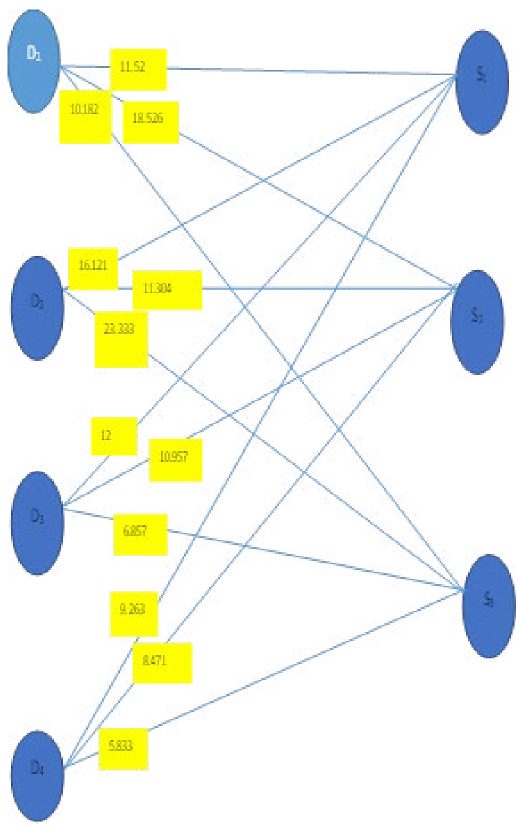
Ex.7

	D_1	D_2	D_3	D_4	Supply
S_1	16 9	19 14	12 12	11 8	16
S_2	22 16	13 10	9 14	8 9	28
S_3	14 8	28 20	8 6	7 5	12
Damand	10	15	17	14	

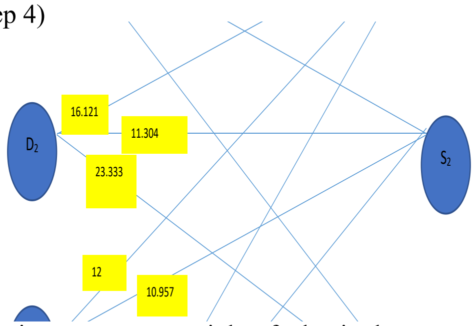
Step 1)

	D_1	D_2	D_3	D_4	Supply
S_1	11.52	16.121	12	9.263	16
S_2	18.526	11.304	10.957	8.471	28
S_3	10.182	23.333	6.857	5.833	12
Damand	10	15	17	14	

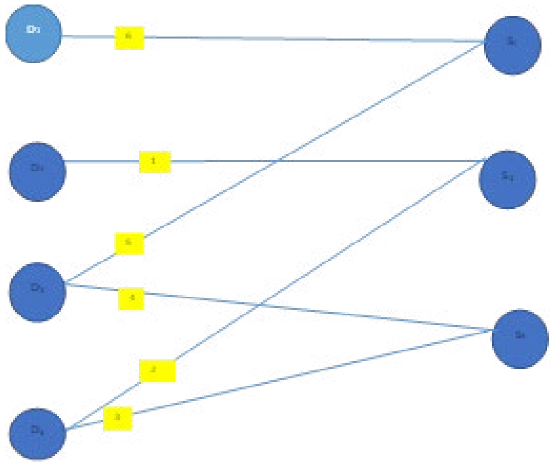
Step 3)



Step 4)



Maximum average weight of edge in the vertex D_2 (16.973). We construct spanning tree using the Depth first search. Starting vertex D_2 and then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step5 and Step 6)

	D_1	D_2	D_3	D_4	Supply
S_1	16 10 9	19 14	12 6 12	11 8	16 (10) (0)
S_2	22 16	13 15 10	9 14	8 13 9	28 (13) (0)
S_3	14 8	28 20	8 11 6	7 1 5	12 (11) (0)
Damand	10 (0)	15 (0)	17 (6) (0)	14 (1) (0)	

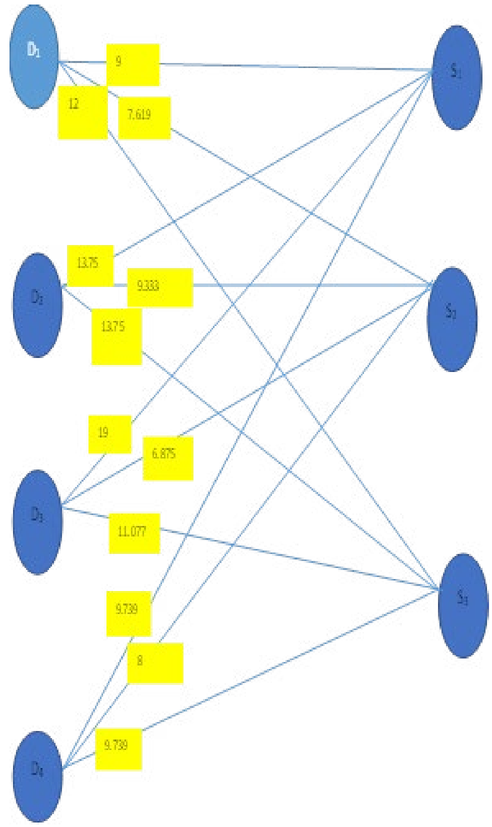
Ex.8

	D_1	D_2	D_3	D_4	Supply
S_1	6 18	10 22	19 19	7 16	35
S_2	5 16	7 14	5 11	6 12	30
S_3	8 24	10 22	8 18	7 15	25
Damand	30	26	19	15	

Step 1)

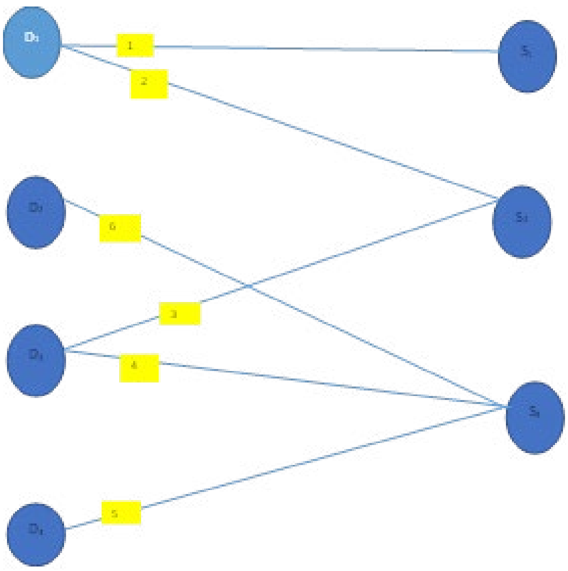
	D_1	D_2	D_3	D_4	Supply
S_1	9	13.75	19	9.739	35
S_2	7.619	9.333	6.875	8	30
S_3	12	13.75	11.077	9.545	25
Damand	30	26	19	15	

Step 3)



Step 4)

Maximum average weight of edge in the vertex S_1 (12.827). We construct spanning tree using the Depth first search. Starting vertex D_2 and then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step5 and Step 6)

	D_1	D_2	D_3	D_4	Supply
S_1	6 30 18	10 5 22	19 19	7 16	35 (5) (0)
S_2	5 16	7 11 14	5 19 11	6 12	30 (11) (0)
S_3	8 24	10 10 22	8 18	7 15 15	25
Damand	30 (0)	26 (16) (5) (0)	19 (0)	15 (0)	

Z_1=(6×30)+(10×5)+(7×11)+(10×10)+(5×19)+(7×15)=607

Z_2=(18×30)+(22×5)+(14×11)+(22×10)+(11×19)+(15×15)=145 8

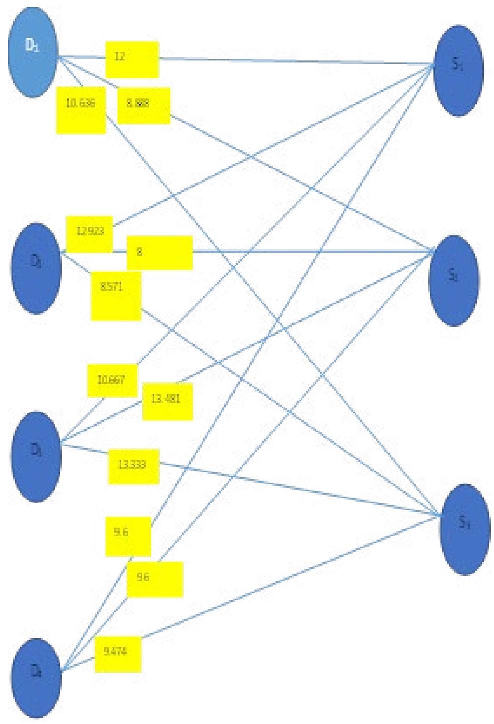
Ex.9

	D_1	D_2	D_3	D_4	Supply
S_1	10 15	14 12	8 16	12 8	15
S_2	8 10	12 6	14 13	8 12	25
S_3	9 13	6 15	15 12	9 10	20
Damand	14	18	12	16	

Step 1)

	D_1	D_2	D_3	D_4	Supply
S_1	12	12.923	10.667	9.6	15
S_2	8.888	8	13.481	9.6	25
S_3	10.636	8.571	13.333	9.474	20
Damand	16	18	12	16	

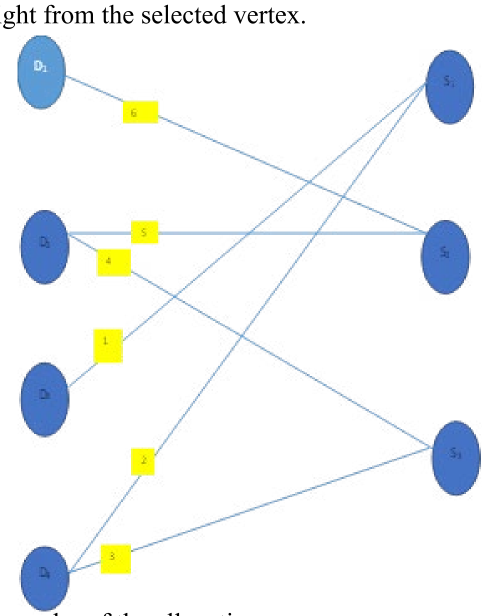
Step 3)



Step 4)

Step 3)

Maximum average weight of edge in the vertex D_3 (12.494). We construct spanning tree using the Depth first search. Starting vertex D_2 and then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step5 and Step 6)

	D_1	D_2	D_3	D_4	Supply
S_1	10 15	14 12	8 12 16	12 3 8	15 (3) (0)
S_2	8 14 10	12 11 6	14 13	8 12	8 12
S_3	9 13	6 7 15	15 12	9 13 10	20 (7) (0)
Damand	14 (0)	18 (11) (0)	12 (0)	16 (13) (0)	

$Z_1=(8\times14)+(12\times11)+(6\times7)+(8\times12)+(12\times3)+(9\times113)=535$
$Z_2=(10\times14)+(6\times11)+(15\times7)+(16\times12)+(8\times3)+(10\times13)=657$

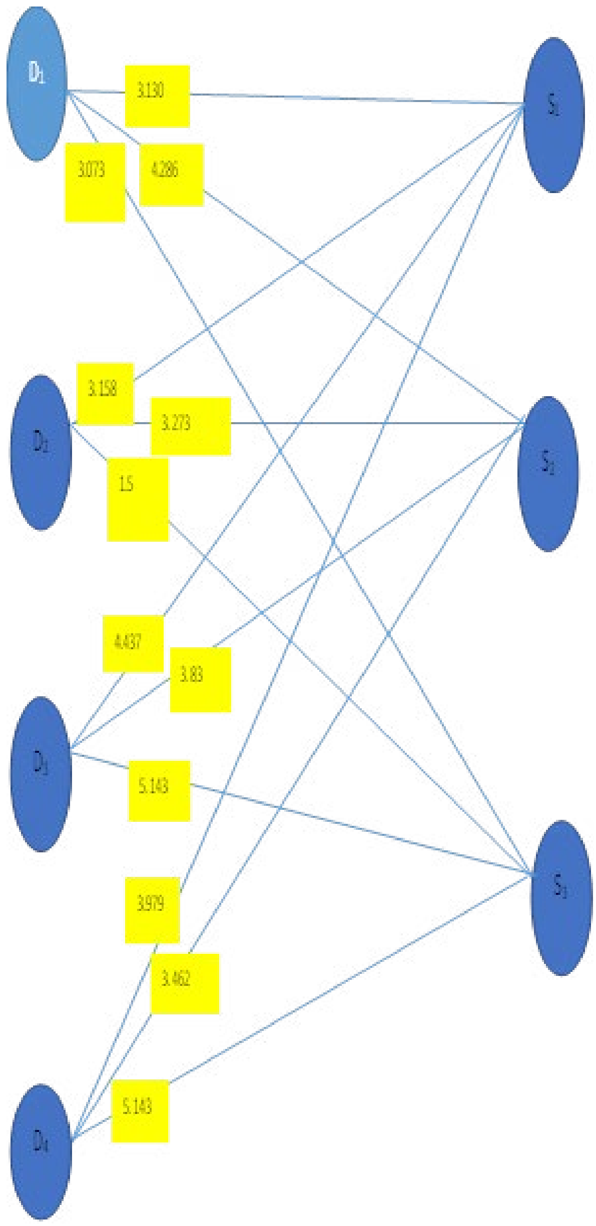
Ex.10

	D_1	D_2	D_3	D_4	Supply
S_1	3 2 8	2 5 4	5 7 3	7 9 2	10
S_2	4 4 5	3 4 3	3 5 4	5 6 2	20
S_3	2 3 7	1 2 2	4 6 6	3 8 8	40
Damand	15	15	20	20	

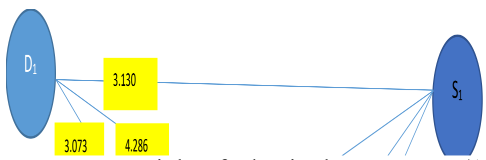
Step 1)

	D_1	D_2	D_3	D_4	Supply
S_1	3.130	3.158	4.437	3.979	15
S_2	4.286	3.273	3.83	3.462	25
S_3	3.073	1.5	5.143	5.143	20
Damand	16	18	12	16	

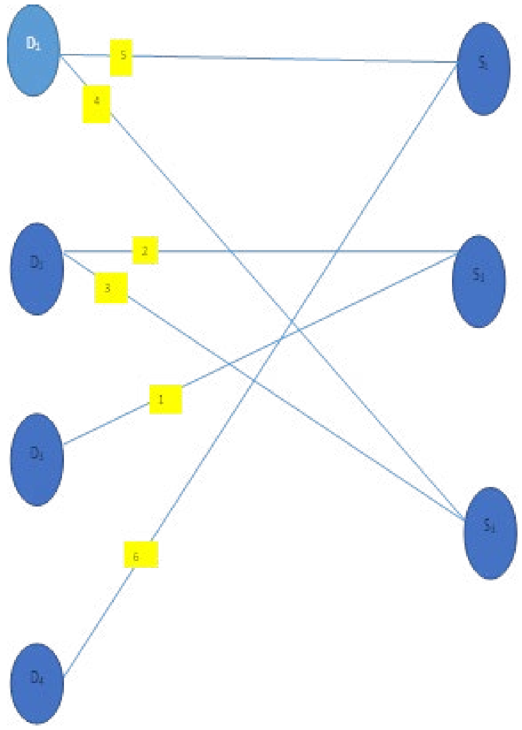
Step 3)



Step 4)



Maximum average weight of edge in the vertex D_3 (12.494). We construct spanning tree using the Depth first search. Starting vertex D_2 and then Select the next vertex along the edge with the least weight from the selected vertex.



1, 2, 3…. -order of the allocation

Step5 and Step 6)

	D_1	D_2	D_3	D_4	Supply
S_1	3 2 8	2 5 4	5 7 3	7 10 9 2	10 (0)
S_2	4 4 5	3 4 3	3 20 5 4	5 6 2	20 (0)
S_3	2 15 3 7	1 15 2 2	4 6 6	3 10 8 8	40 (25) (10) (0)
Damand	15 (0)	15 (0)	20 (0)	20 (10) (0)	

$Z_1=(2\times15)+(1\times15)+(3\times20)+(7\times10)+(3\times10)=205$
$Z_2=(3\times15)+(2\times15)+(5\times20)+(9\times10)+(8\times10)=345$
$Z_3=(7\times15)+(2\times15)+(4\times20)+(2\times10)+(8\times10)=315$

Result and Discussion

A figurative representation of some efficient solutions obtained for multi-objective transportation problems through our proposed approach and algorithms applied by other researchers is shown below. It is shown analytically below using tables and bar Charts.

Ex.no	Table Representations	Bar charts

1

method	Z_1	Z_2
Lakhveer Kaur et al [2]	40	55
Proposed Method	40	55

Table no: 02

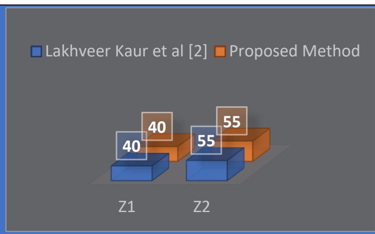


Figure no: 02

2

Table no: 03

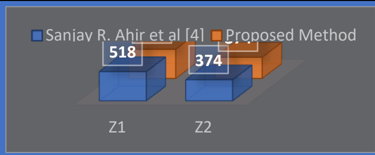


Figure no:03

3

method	Z_1	Z_2
K. Bharathi et al [1]	162	169
Proposed Method	176	175

Table no: 04

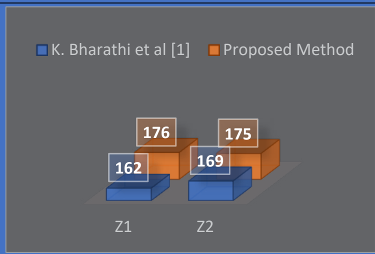


Figure no: 04

4

method	Z_1	Z_2
Sanjay R. Ahir et al [4]	1902	1198
George Osuj [5]	1898	1286
Proposed Method	1902	1198

Table no: 05

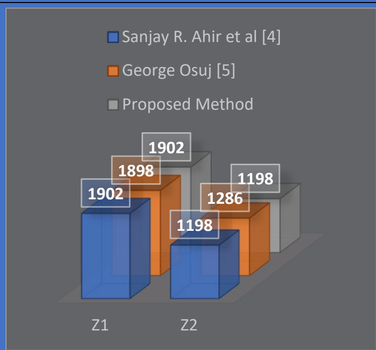


Figure no: 05

5

method	Z_1	Z_2	Z_3
Mohammad Asim Noman et al [3]	127	104	76
Sanjay R. Ahir et al [4]	129	97	83
M. Afwat A.E.	157	72	86

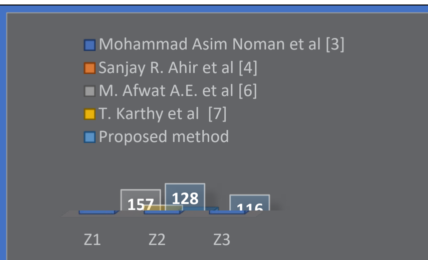


Figure no:06

6

method	Z_1	Z_2
Khilendra singh & Sanjeev Ranjan [15]	550	470
Proposed method	628	430

Table no:07

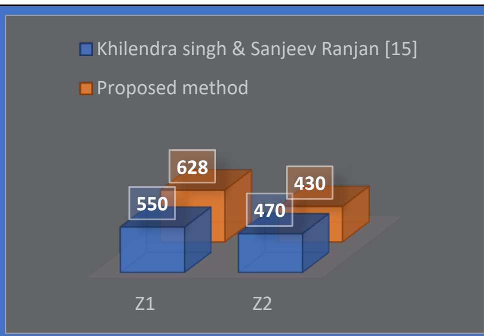


Figure no: 07

7

method	Z_1	Z_2
M.A Mujeer Khan & Syed javed Kabeer [16]	627	491
Proposed method	626	500

Table no: 08

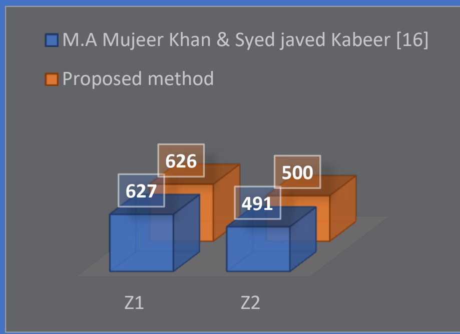


Figure no: 08

Conclusions

Transportation problems are based on minimizing transportation costs by controlling supply and demand constraints. However, in the industry, it is not enough to consider only the transportation cost. Therefore, the researchers formulated the multi-objective transportation problem, in which they considered how to increase the profit of the business through several objectives, such as minimizing time and minimizing energy consumption. In this paper, we developed a new algorithm to find an efficient solution to the multi-objective transportation problem. The efficiency of this method has been shown through the numerical example provided by the algorithm we have prepared. In addition, the solutions obtained to multi-objective transportation problems through our proposed approach and algorithms implemented by other researchers are illustrated [11-19]. In addition, it can be observed that this method can be easily understood and a more efficient solution can be obtained by using a minimum number of steps.

References

1. Bharathi, K., & Vijayalakshmi, C. (2016). Optimization of multi-objective transportation problem using evolutionary algorithms. Global journal of pure and applied mathematics, 12(2), 1387-1396.
2. Kaur, L., Rakshit, M., & Singh, S. (2018). A new approach to solve Multi-objective Transportation Problem. Applications and Applied Mathematics: An International Journal (AAM), 13(1), 10.
3. Masoomi, E., & Rezaei-Moghaddam, K. (2021). Factors affecting entrepreneurial opportunity recognition and exploitation in the context of the Iranian agriculture sector. Iran Agricultural Research, 40(1), 19-36.
4. S. C. D. Barfiwala, "Solution of Multi Objective Transportation Problem."
5. Osuji, G. A., Okoli Cecilia, N., & Opara, J. (2014). Solution of multi-objective transportation problem via fuzzy programming algorithm. Science journal of applied mathematics and statistics, 2(4), 71-77.
6. AE, M. A., Salama, A. A. M., & Farouk, N. (2018). A New Efficient Approach to Solve Multi-Objective Transportation Problem in the Fuzzy Environment (Product approach). International Journal of Applied Engineering Research, 13(18), 13660-13664.
7. Karthy, T., & Ganesan, K. (2018). Multi Objective Transportation Problem-Genetic Algorithm Approach. International Journal of Pure and Applied Mathematics, 119(9), 343-350."
8. Hitchcock, F. L. (1941). The distribution of a product from several sources to numerous localities. Journal of mathematics and physics, 20(1-4), 224-230."
9. Verma, R., Biswal, M. P., & Biswas, A. (1997). Fuzzy programming technique to solve multi-objective transportation problems with some non-linear membership functions. Fuzzy sets and systems, 91(1), 37-43.
10. Hannan, E. L. (1981). Linear programming with multiple fuzzy goals. Fuzzy sets and systems, 6(3), 235-248.

11. Li, L., & Lai, K. K. (2000). A fuzzy approach to the multi-objective transportation problem. Computers & Operations Research, 27(1), 43-57.
12. Zangiabadi, M., & Maleki, H. (2007). Fuzzy goal programming for multiobjective transportation problems. Journal of applied mathematics and Computing, 24(1), 449-460.
13. R. J. Wison. (1996). Introduction to graph theory Fourth edtion.
14. Ogbi, M. S. Z. (2012). A mathematical property of the harmonic mean. The 6th international days of statistics and economics. Prague University of Economics and Business, 873-877."
15. Singh, K., & Rajan, S. (2019). Matrix Maxima Method to Solve Multi-objective Transportation Problem with a Pareto Optimality Criteria.
16. M. A. Mujeeb khan ands.j.kabeer, "Multi-Objective Transportation Problem under Fuzziness with S-type Membership Function."
17. K. K. Jain, R. Bhardwaj, and S. Choudhary. (2019). "A Multi-Objective Transportation Problem Solve By Laxicographic Goal Programming".
18. Doke, D. M., & Jadhav, V. A. (2015). A Solution to three objective transportation problem using fuzzy compromise programming approach. International journal of modern sciences and engineering technology, 2(9), 09-13."
19. Sheikhi, A., Karbassi, S. M., & Bidabadi, N. (2018). A new method for solving bi-objective fractional transportation problems. J. Biochem. Technol, 9, 14-20".