Mathematics in Computer Science manuscript No.
(will be inserted by the editor)

An optimal new method to solve an Assignment problem

Received: date / Accepted: date

Abstract The Assignment problem is introduced as the Maximum weighted bipartite matching problem. Also, this problem nominates as a Minimization problem that comprises a set of positive edge cost instead of edge weight where is at least as large as the maximum of the weight of edge [1]. It should be considered that classical solution to the Assignment problem is proposed which is called Hungarian method[1]. This algorithm and other methods like Simplex method [2], Enumeration method [3] and Transportation method[2] for solving the Assignment problem assures the prior existence of a matrix of edge weights or costs and consequently the problem is addressed with related method. The problem it self may change during the procedure of computing solution. Also new nodes may be added or deleted from related graph or even weight of edges could change.

The Assignment problem also has miscellaneous types such as: Bottleneck Assignment problem [4], Unbalanced Assignment problem [5], Minimum deviation Assignment problem [6], Lexicographic bottleneck Assignment problem [7], $\sum_k$-Assignment problem, Semi Assignment problem [8], Categorized Assignment problem [10], Multi-criteria Assignment problem [10], Fractional Assignment problem [10], Assignment problem with side constraints [10], Quadratic Assignment problem [10]. In this paper a new optimal method to solve an Assignment problem is presented which has less time and calculation complexity in comparison with other classical methods.

Keywords Assignment problem · Maximum weighted matching problem · Linear programming · Transportation problem · Enumeration method · Hungarian algorithm

Shima Zarei
UIVERSITA DEGLI STUDI DI SIENA
Rettorato,Via Banchi di Sotto 55,53100,Siena,ITALIA
Tel.: 0577 232111
E-mail: shima.zarei@student.unisi.it
ORCID ID: 0000-0003-1098-6152

1 Introduction

The Assignment problem is defined as minimization Linear programming problem [9]. Also, Assignment problem is expressed in mathematical form[10]:

$$\sum_{i\in A}\sum_{j\in T} c_{ij}x_{ij} \tag{1}$$

Subject to the constraint :

$$\sum_{j\in T} x_{ij} = 1 \quad , \quad \text{for} \quad j = 1, ..., n, \tag{2}$$

$$\sum_{i\in A} x_{ij} = 1 \quad , \quad \text{for} \quad i = 1, ..., n, \tag{3}$$

$$x_{ij} \geq 0 \tag{4}$$

In these equations variable c_{ij} refers to weight or cost of performance related to tasks and x_{ij} also indicates the Assignment of agent i to task j, taking value 1 if the Assignment is done and 0 otherwise [10].

This problem can be defined in form of a table as well. Suppose there are m jobs and n devices which are available, if the cost of doing jth work by ith person is c_{ij}, then all costs regarding to this issue can be demonstrated as a table 1 :

Devices/Jobs	1	2	3	...j	...m
1	c_{11}	c_{12}	c_{13}	$...c_{1j}$	$...c_{1m}$
2	c_{21}	c_{22}	c_{23}	$...c_{2j}$	$...c_{2m}$
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
.	c_{i1}	c_{i2}	c_{i3}	$...c_{ij}$	$...c_{im}$
n	c_{n1}	c_{n2}	c_{n3}	$...c_{nj}$	$...c_{nm}$

Table 1 Tabular of Assignment problem[10]

2 Solution Methodes:

In order to solve an Assignment problem there are different methods such as Hungarian algorithm [1], Simplex method [11], Enumeration method [3] and Transportation method [2]. These methods are introduced as following:.

2.1) Hungarian Algorithm

Hungarian method [1] is one of the classical solutions of Assignment problem which aim to detect minimum costs for tasks allocation. This algorithm is defined in following steps:

Step 1: Subtract the smallest entry in each row from all the entries of corresponding row.

Step 2: Subtract the smallest entry in each column from all the entries of corresponding column.

Step 3: Draw vertical and horizontal lines through rows and columns in which all the zero entries of the cost matrix are covered with minimum number of lines.

Step 4: (Optimality test): If the minimum number of lines is equal to number of row=n then optimal Assignment is achieved and if minimum number of lines is less than n *go to next step 5.*

Step 5: Subtract minimum value which is not covered by lines from all uncovered values and add it to those values which are covered twice. Then go to step 3 [12].

For more simplicity let's examine this algorithm through an example [13]. Suppose a building firm encompasses 4 cranes each of which has a distance(km) from 4 different construction sites as shown in matrix:

$$Costs = \begin{bmatrix} 90 & 75 & 75 & 80 \\ 35 & 85 & 55 & 65 \\ 125 & 95 & 90 & 105 \\ 45 & 110 & 95 & 115 \end{bmatrix}$$

Step 1: Subrtact minimum value of each row from corresponding row values. So, we have result matrix:

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 5 \\ 0 & 50 & 20 & 30 \\ 35 & 5 & 0 & 15 \\ 0 & 65 & 50 & 70 \end{bmatrix}$$

Step 2: Subtract minimum value of each column from corresponding column values.The result matrix is:

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 50 & 20 & 25 \\ 35 & 5 & 0 & 10 \\ 0 & 65 & 50 & 65 \end{bmatrix}$$

Step 3: Draw line across rows and columns in which all zeros are covered with minimum number of lines. The result matrix is:

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 50 & 20 & 25 \\ 35 & 5 & 0 & 10 \\ 0 & 65 & 50 & 65 \end{bmatrix}$$

Step 4: Perform test for optimality; If number of lines drawn are equal to n *(number of rows) then algorithm is finished. If number of lines are smaller than* n *then go to step 5. Here, number of lines is equal to* $3 \leq n$.

Step 5: Subtract smallest value of uncovered values from each entry not covered by the lines and add it to values which are covered by horizontal and vertical lines. Then check if there is any assignment otherwise repeat step 3.

$$Costs = \begin{bmatrix} 20 & 0 & 5 & 0 \\ 0 & 45 & 20 & 20 \\ 35 & 0 & 0 & 5 \\ 0 & 60 & 50 & 60 \end{bmatrix}$$

In this matrix there is not any assignment. So repeat step 3.Draw lines across zeros.

$$Costs = \begin{bmatrix} 20 & 0 & 5 & 0 \\ 0 & 45 & 20 & 20 \\ 35 & 0 & 0 & 5 \\ 0 & 60 & 50 & 45 \end{bmatrix}$$

Number of lines is 3 which is smaller than n. So subtract minimum value among non-covered values (here is 20) from those values and add it to covered values by lines twice.

$$Costs = \begin{bmatrix} 40 & 0 & 5 & 0 \\ 0 & 25 & 0 & 0 \\ 55 & 0 & 0 & 5 \\ 0 & 40 & 30 & 40 \end{bmatrix}$$

In the result matrix there are 2 option of correct assignment:

Crane1 - Construction site4, Crane2 - Construction site3, Crane3 - Construction site2, Crane4 - Construction site1 (=> overall distance 275 km).

OR

Crane1 - Construction site2, Crane2 - Construction site4, Crane3 - Construction site3, Crane4 - Construction site1 (=> overall distance 275 km) [13].

Example 2: A company wants to offer 5 different job position to 5 employee in which the jobs performance durations(months) are specified in the following matrix:

$$\begin{bmatrix} 11 & 7 & 10 & 17 & 10 \\ 13 & 21 & 7 & 11 & 13 \\ 13 & 13 & 15 & 13 & 14 \\ 18 & 10 & 13 & 16 & 14 \\ 12 & 8 & 16 & 19 & 10 \end{bmatrix}$$

Solution:

Step 1: Subtract minimum value of each row from corresponding row.

$$\begin{bmatrix} 4 & 0 & 3 & 10 & 3 \\ 6 & 14 & 0 & 4 & 6 \\ 0 & 0 & 2 & 0 & 1 \\ 8 & 0 & 3 & 6 & 4 \\ 4 & 0 & 8 & 11 & 2 \end{bmatrix}$$

Step 2: Subtract minimum value of each column from corresponding column.

$$\begin{bmatrix} 4 & 0 & 3 & 10 & 2 \\ 6 & 14 & 0 & 4 & 5 \\ 0 & 0 & 2 & 0 & 0 \\ 8 & 0 & 3 & 6 & 3 \\ 4 & 0 & 8 & 11 & 1 \end{bmatrix}$$

Step 3: Draw line through rows and columns which contains zero:

$$\begin{bmatrix} 4 & 0 & 3 & 10 & 2 \\ 6 & 14 & 0 & 4 & 5 \\ 0 & 0 & 2 & 0 & 0 \\ 8 & 0 & 3 & 6 & 3 \\ 4 & 0 & 8 & 11 & 1 \end{bmatrix}$$

Subtract minimum value which is not covered by lines from those values(here is 1) and add this minimum value to values that are covered by lines twice. Check if there is an Assignment, otherwise repeat step 3 again:

$$\begin{bmatrix} 3 & 0 & 2 & 9 & 1 \\ 5 & 15 & 0 & 4 & 5 \\ 0 & 1 & 2 & 0 & 0 \\ 7 & 0 & 2 & 5 & 2 \\ 3 & 0 & 7 & 10 & 0 \end{bmatrix}$$

In matrix there is not any assignment. In this iteration minimum value of uncovered rows and columns is 2 and subtract it from uncovered values and add it to covered values by lines.

$$\begin{bmatrix} 1 & 0 & 0 & 7 & 1 \\ 5 & 17 & 0 & 4 & 7 \\ 0 & 3 & 2 & 0 & 2 \\ 5 & 0 & 0 & 3 & 2 \\ 1 & 0 & 5 & 8 & 0 \end{bmatrix}$$

As matrix showes there is not any Assignment so repeat step 3 with minimum value equal to 1 .

$$\begin{bmatrix} 0 & 0 & 1 & 6 & 0 \\ 4 & 17 & 0 & 3 & 6 \\ 0 & 4 & 3 & 0 & 2 \\ 4 & 0 & 0 & 2 & 1 \\ 1 & 1 & 6 & 8 & 0 \end{bmatrix}$$

In the result matrix there is an Assignment which specifies by green color. Hence, in original matrix the values related to this highlighted zeros are 11, 7, 13, 10, 10 and minimum total time for performing jobs by employees are:

$11 + 7 + 13 + 10 + 10 = 51$

In the innovation section 3 this problem is solved in optimal way which reduced time and calculation complexity.

Example 3: A company tends to assign 5 project to 5 engineer for supervising on them. The time of projects supervision by each engineer is shown in matrix:

$$\begin{bmatrix} 15 & 5 & 9 & 8 & 17 \\ 11 & 16 & 19 & 6 & 3 \\ 9 & 18 & 5 & 20 & 25 \\ 17 & 10 & 8 & 21 & 20 \\ 7 & 14 & 12 & 9 & 10 \end{bmatrix}$$

Then problem is how assign projects to engineers that minimum time spends for projects supervision?

Solution:

Step 1: Subtract minimum value of each row from corresponding row:

$$\begin{bmatrix} 10 & 0 & 4 & 3 & 12 \\ 8 & 13 & 16 & 3 & 0 \\ 4 & 13 & 0 & 15 & 20 \\ 9 & 2 & 0 & 13 & 12 \\ 0 & 7 & 5 & 2 & 3 \end{bmatrix}$$

Step 2: Subtract minimum value of each column from corresponding column :

$$\begin{bmatrix} 10 & 0 & 4 & 1 & 12 \\ 8 & 13 & 16 & 1 & 0 \\ 4 & 13 & 0 & 13 & 20 \\ 9 & 2 & 0 & 11 & 12 \\ 0 & 7 & 5 & 0 & 3 \end{bmatrix}$$

Step 3: Draw minimum number of lines through rows and columns in which covers all zero values :

$$\begin{bmatrix} 10 & 0 & 4 & 1 & 12 \\ 8 & 13 & 16 & 1 & 0 \\ 4 & 13 & 0 & 13 & 20 \\ 9 & 2 & 0 & 11 & 12 \\ 0 & 7 & 5 & 0 & 3 \end{bmatrix}$$

Subtract minimum value which is not covered by lines from all uncoverd values and add it to values which are covered by lines twice and here it is 1 :

$$\begin{bmatrix} 10 & 0 & 6 & 1 & 12 \\ 8 & 13 & 18 & 1 & 0 \\ 2 & 11 & 0 & 11 & 218 \\ 7 & 0 & 0 & 9 & 10 \\ 0 & 7 & 7 & 0 & 3 \end{bmatrix}$$

As matrix shows there is not any assignment and repeat step 3 .

$$\begin{bmatrix} 10 & 0 & 6 & 1 & 12 \\ 8 & 13 & 18 & 1 & 0 \\ 2 & 11 & 0 & 11 & 18 \\ 7 & 0 & 0 & 9 & 10 \\ 0 & 7 & 7 & 0 & 3 \end{bmatrix}$$

Then subtract minimum value of uncovered values from those values and add it to one which are covered twice and check it for optimality. After subtracting minimum uncovered *value which is equal to 1 from those values and adding it to covered twice values, the result matrix is:*

$$\begin{bmatrix} 9 & 0 & 6 & 0 & 11 \\ 8 & 14 & 19 & 1 & 0 \\ 1 & 11 & 0 & 10 & 17 \\ 6 & 0 & 0 & 8 & 9 \\ 0 & 8 & 8 & 0 & 3 \end{bmatrix}$$

As it is highlighted in matrix there is a correct assignment. The actual value from origin matrix which is corresponded to those highlighted values are:

8+3+5+10+7=33

Therefore, the total cost of this problem is equal to 33.

2.2) Simplex Method

Simplex method [11] is used for solving Linear programming problems in standard form where objective function was to be maximized and it is extended to the form where objective function is to be minimized. It is noted that in case of solving an Assignment problem objective function must be minimized. This procedure is written as below: A minimization problem is in standard form if the objective function:

$W = c_1x_1 + c_2x_2 + ... + c_nx_n$ *is to be minimized, subject to the constraints:*

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n \geq b_1 \tag{5}$$

$$a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \geq b_2 \tag{6}$$

$$\vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n \geq b_m \tag{7}$$

where $x_i \geq 0$ *and* $b_i \geq 0$ *. After adding slack variables, the corresponding system of constraint equations are:*

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n + s_1 \qquad = b_1 \tag{8}$$

$$a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \qquad + s_2 = b_2 \tag{9}$$

$$\vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n \qquad + s_m = b_m \tag{10}$$

where $s_i \geq 0$ *[11].*

For instance, the objective function is : $z = 4x_1 + 6x_2$ *and constraints are as following:*

$-x_1 + x_2 + s_1 = 11$

$x_1 + x_2 + \quad +s_2 = 27$

$2x_1 + 5x_2 + \qquad +s_3 = 90$

secondly, set up the initial tableau as following 2:

-1	1	1	0	0	11
1	1	0	1	0	27
2	5	0	0	1	90
-4	-6	0	0	0	0

Table 2 cost table [11]

from this initial tabaleau a basic feasible solution is:
$(x_1, x_2, s_1, s_2, s_3) = (0, 0, 11, 27, 90)$

To perform an optimality check for a solution represented by a simplex tableau, look at the enteries in the bottom row of the tableau. If any of the enteries are negative 2 then the current solution is not optimal [11][17].

2.3) Enumeration Method

The Enumeration method [3] for Assignment problem is used for small set of data. In this method all possible Assignments are selected from set of data and then the one with minimum cost is selected as a solution of Assignment problem. For instance we have a matrix of 3 processor and 3 applications and we want to assign each application to each processor to run applications with minimum time. The matrix of data is as following :

$$Times: \begin{bmatrix} 5 & 3 & 2 \\ 6 & 8 & 4 \\ 1 & 9 & 7 \end{bmatrix}$$

The possible Assignment is as following:
5+8+7=20
3+6+7=16
2+8+1=11
5+4+9=18
3+4+1=8
2+9+6=17

Therefore, the solution is to assign application 1 to processor 2 and application 2 to processor 3 and application 3 to processor 1 with total time equal to 8 minutes [3].

If this method apply on example 3 of Hungarain method 2.1 the procedure and result are as following:

$$\begin{bmatrix} 15 & 5 & 9 & 8 & 17 \\ 11 & 16 & 19 & 6 & 3 \\ 9 & 18 & 5 & 20 & 25 \\ 17 & 10 & 8 & 21 & 20 \\ 7 & 14 & 12 & 9 & 10 \end{bmatrix}$$

Then it is possible to consider:
$15 + 16 + 5 + 21 + 10 = 67$
$15 + 19 + 20 + 10 + 10 = 74$
$15 + 6 + 5 + 10 + 10 = 45$
$15 + 6 + 25 + 8 + 14 = 68$
$15 + 3 + 20 + 8 + 14 = 60$

$15+9+20+5+16=65$
$15+12+21+18+3=69$
$5+11+5+21+10=52$
$5+19+9+20+9=62$
$5+6+5+17+10=43$
.
.
.
$8+3+5+10+7=33$

Therefore, after consideration of all possible assignmenst the one which has minimum cost is last one with total cost equal to 33. As it is indicated this method has more time complexity than Hungarian method. Nevetheless, it presents equal answer on each assignment problem as other solutions.

2.4)Transportation Method

Transportation method is one type of linear programming problem that is solved by a simplified version of the simplex method [14]. This problem is concerned with finding the minimum cost of transporting a single commodity from a given number of sources to number of destinations. These type of problems can be solved by general network methods.

In the following equations X_{ij} indicates the quantity of transported goods from source i to destination j. The cost associated with this movement is $X = C_{ij}X_{ij}$. The cost of transporting the commodity from source i to all destinations is given by [15]:

$$\sum_{i=1}^{n} C_{ij}X_{ij} = C_{i1}X_{i1} + C_{i2}X_{i2} + ... + C_{in}X_{in} \tag{11}$$

Therefore, total cost of transportation is:

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{12}$$

$$= c_{11}x_{11} + c_{12}x_{12} + ... + c_{1n}x_{1n} + \tag{13}$$

$$c_{21}x_{21} + c_{22}x_{22} + ... + c_{2n}x_{2n} + \tag{14}$$

$$c_{m1}x_{m1} + c_{m2}x_{m2} + ... + c_{mn}x_{mn} \tag{15}$$

[15]

For purpose of transportation cost minimization, the following problem must be solved [15]:

$$Minimize \quad Z = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}, \tag{16}$$

$$Subject \quad to \quad \sum_{j=1}^{n} x_{ij} \le a_i \quad for \quad i = 1, ..., m \tag{17}$$

$$\sum_{i=1}^{m} x_{ij} \ge b_j \quad for \quad j = 1, ..., n \tag{18}$$

$$where \quad x_{ij} \geq 0 \quad for \qquad i = 1, ..., m \quad and \quad j = 1, ..., n \tag{19}$$

In terms of balanced transportation problem for each constraints must have following equation:

$$\sum_{j=1}^{n} x_{ij} = a_i \quad for \quad i = 1, ..., m, \tag{20}$$

$$\sum_{i=1}^{m} x_{ij} = b_j \quad for \quad j = 1, ..., n \tag{21}$$

[15] The transportation problem will have feasible solution if and only if:

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \tag{22}$$

A transportation model in which the total supplies and total demands are unequal is called unbalanced transportation problem. It is always possible to balance an unbalanced transportation problem [15].

Example: Powerco has 3 electric power plant that supply the needs of 4 cities. Each power plant can supply the following numbers of kilowatt-hours (kwh) of electricity:power plant 1-35 million;power plant 2-50 million;power plant 3-40 million, table2.4. The peak power demands in these cities, which occurs at the same time (2pm), are as following (in kwh): city 1-45 million; city 2-20 million; city 3-30 million; city 4-30 million. The costs of sending 1 million kwh of electricity from power plant to city depend on the distance the electricity must travel. Formulate an LP to minimize the cost of meeting each city's peak power demand [15].

From	city1	city2	city3	city4	supply(kwh)
Plant1	$8	$6	$10	$9	$35
Plant2	$9	$12	$13	$7	$50
Plant3	$14	$9	$16	$5	$40
Demand(kwh)	45	20	30	30	

Table 3 cost of transportation [15]

Decision Variables:
x_{ij} *# of (million) kwh produced at power palnt $\underline{i}$ and send to city $\underline{j}$.*
Constraints: Supply constraints and Demand constraints.
Then problem formulates as below:
Minimize total shipping cost:
Min $8x_{11} + 6x_{12} + 10x_{13} + 9x_{14} +$
$9x_{21} + 12x_{22} + 13x_{23} + 7x_{24} +$
$14x_{31} + 9x_{32} + 16x_{33} + 5x_{34}$
Supply constraints:
$x_{11} + x_{12} + x_{13} + x_{14} \leq 35$
$x_{21} + x_{22} + x_{23} + x_{24} \leq 50$
$x_{31} + x_{32} + x_{33} + x_{34} \leq 40$
Demand constraints
$x_{11} + x_{21} + x_{13} \geq 45$

$x_{12} + x_{22} + x_{32} \geq 20$
$x_{13} + x_{23} + x_{33} \geq 30$
$x_{14} + x_{24} + x_{34} \geq 30$
$x_{ij} \geq 0$ *(i= 1,2,3 ; j=1,2,3,4)*
Then optimal solution is :
$z = 1020, x_{12} = 10, x_{13} = 10, x_{21} = 45, x_{23} = 5, x_{32} = 10, x_{34} = 34$ *, all other variables are equal to zero [15].*

3 Innovated Technique

In this paper a New and Optimal method for solving Assignment problem is presented in which can detect minimum cost or time or path for an Assignment problem. This optimal algorithm defines as below:

Algorithm : Optimal Method
*Procedure Optimal algorithm(M: n*n Matrix of integers)*
for $i = 0$ ***to*** n ***do***
  begin
  u_i*= Minimum integer in row i of M*
  for $j = 1$ ***to*** n ***do***
    M̂= c_{ij} *-* u_i
  end for
end for
for $j = 1$ ***to*** n ***do***
  begin
  v_j *= Minimum integer in column j of M̂*
  for $i = 1$ ***to*** n ***do***
    M̂ = $\hat{M}_{ij}$*-* v_j
  end for
end for{*M̂ is now the reduced Matrix*}
S = a set of non-zero values with maximum sum among all rows with non-zero values on same columns in matrix M̂ .
$h = |S|$
while *M̂ does not contains any minimized assignment* ***do***
  cover (h in M̂)
  D = Minimum entry in h covered by a line
  for $i = 1$ ***to*** n ***do***
    for $j = 1$ ***to*** n ***do***
      begin
      if h in M̂ is covered then M̂ = M̂ - D
    end for
  end for
end while
for $i = 1$ ***to*** n ***do***
  for $j = 1$ ***to*** n ***do***
    if *M̂*$\in S$ ***then***
      $X_{ij} = 1$
    else
      $X_{ij} = 0$

end if
end for
end for

This new algorithm is defined in more simple way as following:

Step 1 :Subtract minimum value of each row from corresponding row.

Step 2 :Subtract minimum value of each column from corresponding column. Then check if there is correct Assignment, otherwise go to next step.

Step 3: Select rows with non-zero values on same columns in which those values have maximum summation among all summation of rows with non-zero values on same columns. Then subtract minimum value of those values from all of them. Finally check if there is a correct Assignment, otherwise repeat step 3 until reach a correct Assignment.

3.1 Optimal algorithm in terms of bipartit graph

A complete bipartit graph is denoted by G = (U , V ; E) with n devices vertices (U) and n jobs vertices (V) and each edge has non-negative costs C(i , j) and the aim is to find a perfect matching with minimum cost. Therefore, function $y : (\bar{U} \cup V) \Rightarrow R$ *is a potential if* $y(i)+y(j) \leq c(i,j)$ *for each* $i \in U$ *and* $j \in V$. *The value of potential y is* $\sum_{v \in U \cup V} y(V)$. *It can be detected that cost of each perfect matching is at least the value of each potential. The new optimal algorithm tries to find perfect matching. This algorithm detects the perfect matching of tight edges [16].*

For more simpilicity draw the Bipartit graph [16] corresponding to matrix with consideration of correponding cost for each edge. Then U nodes refers to rows and V nodes refers to columns and edges weights refers to values of data from nodes U to nodes V. Then algorithm defines as below:

Step 1: Subtract minimum value of edges which goes out from U nodes from all of corresponding edges.

Step 2: Subtract minimum value of edges which goes out from V nodes from all of corresponding edges.

Step 3: If it is perfect matching then problem is finished. Otherwise, select edges from U to V in which they have termination on same V nodes with maximum sum of edges weights among those one with same termination on V nodes.

Step 4: Subtract minimum value of selected edges in previous step from all of them. Then check if it is perfect matching. Otherwise, repeat step 3 until reach perfect matching.

3.2 Time Complexity of Optimal algorithm

Time complexity of Innovated algorithm is less than Hungarian algorithm by Kuhn [1] in some problems, because this algorithm takes less number of iterations to solve the assignment problem than Hungarian method. So it has lower time complexity in comparison with Hungarian method. It expresses in implemenation part 3.3 through some examples.

3.3 Implementation of Optimal algorithm on some examples

Let's explain this algorithm more explicit through some examples of Assignment problem;

The matrix of costs which is corresponded to distances of 4 cranes from 4 construction sites is presented as below:

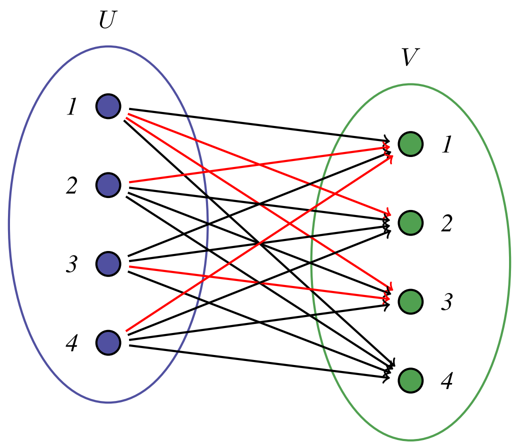
$$Costs = \begin{bmatrix} 90 & 75 & 75 & 80 \\ 35 & 85 & 55 & 65 \\ 125 & 95 & 90 & 105 \\ 45 & 110 & 95 & 115 \end{bmatrix}$$

Solution:

Step 1: Subtract minimum value of each row from its values .

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 5 \\ 0 & 50 & 20 & 30 \\ 35 & 5 & 0 & 15 \\ 0 & 65 & 50 & 70 \end{bmatrix}$$

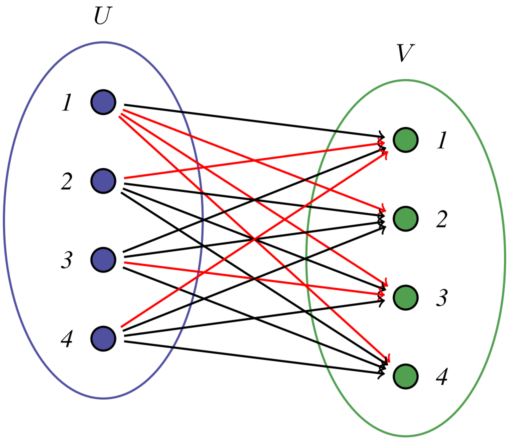
The bipartit graph of cost edges is shown as below that red edges implies zero cost edges:



Step 2: Subtract minimum value of each column from its values .

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 50 & 20 & 25 \\ 35 & 5 & 0 & 10 \\ 0 & 65 & 50 & 65 \end{bmatrix}$$

As matrix shows there is not any Assignment and bipartit graph of edges are as below in which zero edges are highlighted with red color:

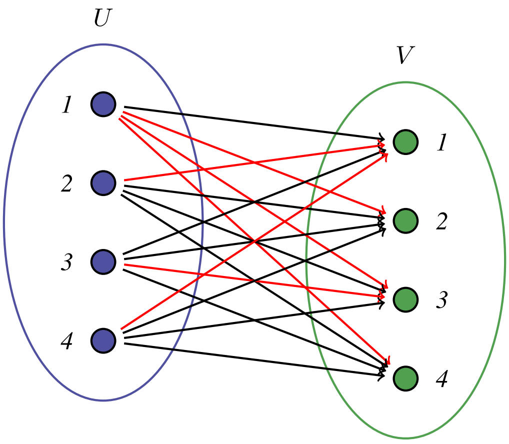


In this bipartit graph also there is not perfect matching. Therefore, go to step 3.

Step 3: In this step must choose rows 2 and 4 and columns 2, 3, 4 because these rows and columns have maximum sum of values among all rows that have non-zero values on same columns. These values are highlighted in matrix as below:

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 50 & 20 & 25 \\ 35 & 5 & 0 & 10 \\ 0 & 65 & 50 & 65 \end{bmatrix}$$

and the bipartit graph of matrix is as following:



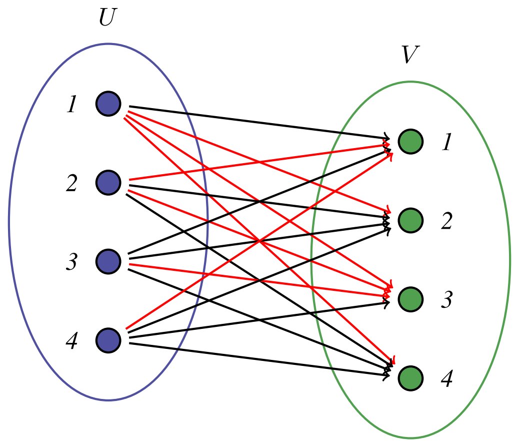
Then subtract minimum value of these values that is 20 from related values. Therefore, the cost matrix is as below :

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 30 & 0 & 5 \\ 35 & 5 & 0 & 10 \\ 0 & 45 & 30 & 45 \end{bmatrix}$$

In this step there is not any Assignment. So repeat step 3: Select rows 2, 3, 4 and columns 2, 4 from cost matrix which have maximum sum of costs among rows which has non-zero value on same columns .

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 30 & 0 & 5 \\ 35 & 5 & 0 & 10 \\ 0 & 45 & 30 & 45 \end{bmatrix}$$

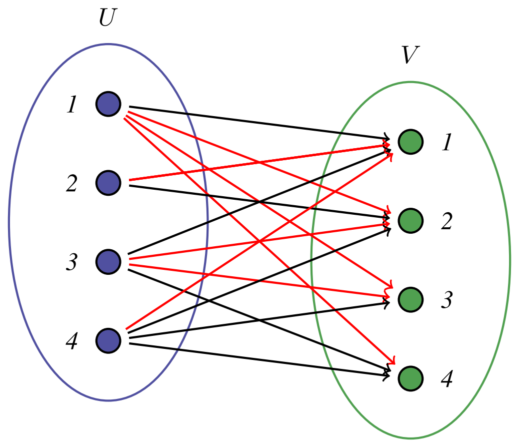
and draw the bipartit graph of matrix :



Then subtract 5 which is minimum value of selected rows and columns from all values of corresponding rows and columns. As you can see in bipartit graph nodes 2, 3, 4 of U set have edges with same termination on nodes 2, 4 of V set and have maximum some of cost among all edges from U set to V set. Consequently the cost matrix is as below:

$$Costs = \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 25 & 0 & 0 \\ 35 & 0 & 0 & 5 \\ 0 & 40 & 30 & 40 \end{bmatrix}$$

Also bipartit graph corresponding to matrix is as following:



In this matrix we can find Assignment explicitly. So the solution with total cost of transportation with cosidearation of original cost matrix is:

80+55+95+45=275

or

75+65+90+45=275

Example 2: A company offers 5 different job position to 5 employee in which the jobs performance durations are specified in the following matrix:

$$\begin{bmatrix} 11 & 7 & 10 & 17 & 10 \\ 13 & 21 & 7 & 11 & 13 \\ 13 & 13 & 15 & 13 & 14 \\ 18 & 10 & 13 & 16 & 14 \\ 12 & 8 & 16 & 19 & 10 \end{bmatrix}$$

Step 1: Subtract minimum value of each row from corresponding values.

$$\begin{bmatrix} 4 & 0 & 3 & 10 & 3 \\ 6 & 14 & 0 & 4 & 6 \\ 0 & 0 & 2 & 0 & 1 \\ 8 & 0 & 3 & 6 & 4 \\ 4 & 0 & 8 & 11 & 2 \end{bmatrix}$$

Step 2: Subtract minimum value of each column from corresponding values.

$$\begin{bmatrix} 4 & 0 & 3 & 10 & 2 \\ 6 & 14 & 0 & 4 & 5 \\ 0 & 0 & 2 & 0 & 0 \\ 8 & 0 & 3 & 6 & 3 \\ 4 & 0 & 8 & 11 & 1 \end{bmatrix}$$

Step 3: Select rows 1, 2, 4, 5 and columns 1, 4, 5 which have maximum sum of values among rows with non-zero values on same columns:

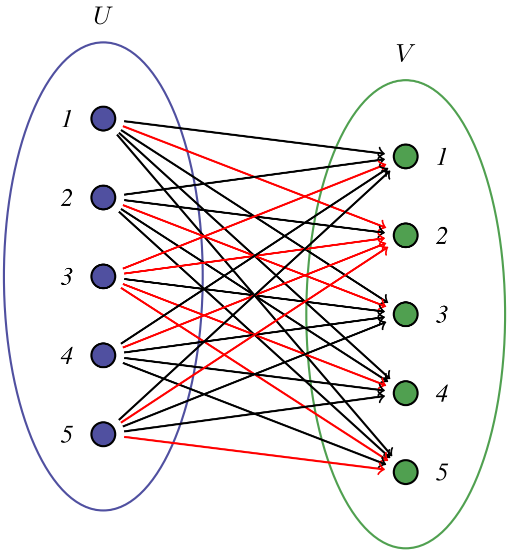
$$\begin{bmatrix} 4 & 0 & 3 & 10 & 2 \\ 6 & 14 & 0 & 4 & 5 \\ 0 & 0 & 2 & 0 & 0 \\ 8 & 0 & 3 & 6 & 3 \\ 4 & 0 & 8 & 11 & 1 \end{bmatrix}$$

Now subtract minimum value of these values from all of them. Then, check if there is an Assignment, otherwise repeat step 3.

Therefore, the matrix after calculation is:

$$\begin{bmatrix} 3 & 0 & 3 & 9 & 1 \\ 5 & 14 & 0 & 3 & 4 \\ 0 & 0 & 2 & 0 & 0 \\ 7 & 0 & 3 & 5 & 2 \\ 3 & 0 & 8 & 10 & 0 \end{bmatrix}$$

Now draw the bipartit graph corresponding to elements of this matrix and highlight zero edges:



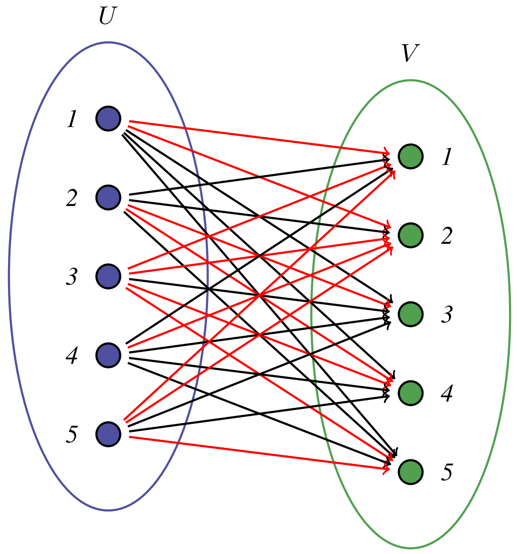
Because in this matrix there is not any Assignment, repeat step 3. So, select rows 1, 2, 4, 5 and columns 1, 4 which has maximum sum of values among rows with non-zero values on same columns.

$$\begin{bmatrix} 3 & 0 & 3 & 9 & 1 \\ 5 & 14 & 0 & 3 & 4 \\ 0 & 0 & 2 & 0 & 0 \\ 7 & 0 & 3 & 5 & 2 \\ 3 & 0 & 8 & 10 & 0 \end{bmatrix}$$

Then subtract 3 that is minimum of these values from them. Consequently, the matrix of values is as below:

$$\begin{bmatrix} 0 & 0 & 3 & 6 & 1 \\ 5 & 14 & 0 & 0 & 4 \\ 0 & 0 & 2 & 0 & 0 \\ 4 & 0 & 3 & 2 & 2 \\ 0 & 0 & 8 & 7 & 0 \end{bmatrix}$$

and corresponding bipartit graph is such as below:



In this matrix there is a correct Assignment which values related to those highlited zeros in original matrix are as following with total amount of cost:

$11 + 13 + 10 + 10 + 7 = 51$

In comparison with solution of this example by Hungarian Algorithm which has $O(n^4)$ time complexity, this solution earns lower time complexity that is equal to $O(n^3)$. This is because of lower number of iteration of new algorithm on matrix of data to reach the correct assignment in contrast to Hungarian method. Therefore, this new method is useful and optimal because it has lower time complexity in some problem than Hungarian Methods.

Example 3: A company tends to assign 5 project to 5 engineer for supervising on them. The times of projects supervision by each engineer are shown in matrix as below:

$$\begin{bmatrix} 15 & 5 & 9 & 8 & 17 \\ 11 & 16 & 19 & 6 & 3 \\ 9 & 18 & 5 & 20 & 25 \\ 17 & 10 & 8 & 21 & 20 \\ 7 & 14 & 12 & 9 & 10 \end{bmatrix}$$

The problem is how assign projects to engineers that minimum time spends for projects supervision?

Solution:

Step1: Subtract minimum value of each row from related values.

$$\begin{bmatrix} 10 & 0 & 4 & 3 & 12 \\ 9 & 10 & 13 & 3 & 0 \\ 4 & 13 & 0 & 15 & 20 \\ 9 & 2 & 0 & 13 & 12 \\ 0 & 7 & 5 & 2 & 3 \end{bmatrix}$$

Step2: Subtract minimum value of each column from related values.

$$\begin{bmatrix} 10 & 0 & 4 & 1 & 12 \\ 9 & 10 & 13 & 1 & 0 \\ 4 & 13 & 0 & 13 & 20 \\ 9 & 2 & 0 & 11 & 12 \\ 0 & 7 & 5 & 0 & 3 \end{bmatrix}$$

Step3: select non-zero values of rows 3 and 4 which contains non-zero values on same columns among all rows with non-zero values on same columns(are highlighted).

$$\begin{bmatrix} 10 & 0 & 4 & 1 & 12 \\ 9 & 10 & 13 & 1 & 0 \\ 4 & 13 & 0 & 13 & 20 \\ 9 & 2 & 0 & 11 & 12 \\ 0 & 7 & 5 & 0 & 3 \end{bmatrix}$$

Then subtract minimum value of those highlighted value in matrix from those values. The result matrix is as below:

$$\begin{bmatrix} 10 & 0 & 4 & 1 & 12 \\ 9 & 10 & 13 & 1 & 0 \\ 2 & 11 & 0 & 11 & 18 \\ 7 & 0 & 0 & 9 & 10 \\ 0 & 7 & 5 & 0 & 3 \end{bmatrix}$$

In this matrix there is not any correct assignment and repeat step 3. Therefore, select non-zero values of rows 3,4 which has non-zero value on columns 1,4,5 with maximum sum of values among rows with non-zero values on same columns. The result matrix is as following:

$$\begin{bmatrix} 10 & 0 & 4 & 1 & 12 \\ 9 & 10 & 13 & 1 & 0 \\ 2 & 11 & 0 & 11 & 18 \\ 7 & 0 & 0 & 9 & 10 \\ 0 & 7 & 5 & 0 & 3 \end{bmatrix}$$

Then subtract minimum value of those selected values which is equal to 2 from those values. The result matrix is:

$$\begin{bmatrix} 9 & 0 & 4 & 0 & 11 \\ 9 & 10 & 13 & 1 & 0 \\ 1 & 11 & 0 & 10 & 17 \\ 6 & 0 & 0 & 8 & 9 \\ 0 & 7 & 5 & 0 & 3 \end{bmatrix}$$

As it is indicated in previous matrix there is a correct assignment (highlighted values) in this matrix and with refer to original matrix the total cost of this problem is equal to :

$8 + 3 + 5 + 10 + 7 = 33$

Complexity of this new method as a solution for this example is $O(n^4)$ which is equal to complexity of Hungarian method in this case.

In conclusion, Linear programming problem which has several techniques like solutions that are mentioned in this paper has some difficulties for solving Assignment problem such as time complexity and calculation complexity. Therefore, for ease of solving this problem *the innovated method which is presented in this paper is introduced and implemented. By using this new algorithm, Assignment problem is solved in shorter period of time and very precisely for small size and huge data sets as well. Also in a case when previous classical methods are not efficient this new technique is well suitable.*

Acknowledgements *I appreciate all of scientists and authors that are cited in this paper as my scientific references. Thanks for all of their relevant papers and books which guided me to write this paper.*

References

1. , Kuhn, Harold W, The hungarian method for the Assignment problem, 50 Years of Integer Programming 1958-2008, 29–47, Springer, (2010)
2. , Balinski, ML and Gomory, RE, A primal method for the Assignment and transportation problems, Management Science, 10(3):578–593, INFORMS, (1964)
3. , Thomas, Jech, Set Theory: Third Millennium Edition, Springer Monographs in Mathematics, Springer, 642, (2002)
4. , Aggarwal, Vijay and Tikekar, VG and Hsu, Lie-Fern, Bottleneck Assignment problems under categorization, Computers & operations research, 13,1,11–26, Elsevier, (1986)
5. , Kumar, Avanish, A modified method for solving the unbalanced Assignment problems, Applied mathematics and computation, 176,1,76–82, Elsevier, (2006)
6. , Gupta, SK and Punnen, AP, Minimum deviation problems, Operations research letters, 7,4,201–204, Elsevier, (1988)
7. , Burkard, Rainer E and Rendl, Franz, Lexicographic bottleneck problems, Operations Research Letters, 10,5,303–308, Elsevier, (1991)
8. , Volgenant, A, Linear and semi-Assignment problems: a core oriented approach, Computers & Operations Research, 23,10,917–932, Elsevier, (1996)
9. , Ferguson, Thomas S, Linear programming: A concise introduction. http://www. math. ucla. edu/˜tom/LP.pdf, (2002)
10. , Pentico, David W, Assignment problems: A golden anniversary survey, European Journal of Operational Research, 176,2,774–793, Elsevier, (2007)
11. , Ferguson, Robert O and Sargent, Lauren F, Linear programming, McGraw-Hill, (1958)
12. , Munkres, James, Algorithms for the Assignment and transportation problems, Journal of the society for industrial and applied mathematics, 5,1,32–38, SIAM, (1957)
13. , Bruff, Derek, The Assignment problem and the hungarian method, 20,29–47, Notes for Math, (2005)
14. , Heineman, George T and Pollice, Gary and Selkow, Stanley, Algorithms in a nutshell: a practical guide, " O'Reilly Media, Inc.", (2016)
15. , Winston, Wayne L and Venkataramanan, Munirpallam and Goldberg, Jeffrey B, Introduction to mathematical programming, 1, Thomson/Brooks/Cole Duxbury; Pacific Grove, CA, (2003)
16. , Asratian, Armen S and Denley, Tristan MJ and Häggkvist, Roland, Bipartite graphs and their applications, 131, Cambridge University Press, (1998)
17. , Balinski, Michel L, A competitive (dual) simplex method for the Assignment problem, Mathematical Programming, 34,2,125–141, Springer, (1986)