ISSN 2278-9723
DOI:10.31838/ijccts/09.01.06

Research Article

Improved Hungarian Algorithm for Unbalanced Assignment Problems

K. ANNAPURNA[1], A. MOUNIKA YESASWINI[2]
[1,2]VFSTR deemed to be University, Andhra Pradesh, India
Email: arya.anu85@gmail.com[1], mouni.ammanabrolu@gmail.com[2]
Received: 12.07.21, Revised: 17.08.21, Accepted: 10.09.21

ABSTRACT

Hungarian algorithm gives optimum one to one assignment when there are equal number of machines and jobs. For unbalanced assignment problems, prior to solve it, dummy jobs/machines are to be added to convert the unbalanced problem in to a balanced problem. But the jobs which are assigned to dummy machines cannot be served in reality. So to avoid this problem duplication of required number of machines/jobs, that is multiple jobs (machines) are assigned to a single machine (job) is proposed. At most care is taken while selecting the duplicate machines/jobs to minimize the cost of final assignment. In addition to that the proposed algorithm ensures no overloading of particular machine/job. Some researchers have proposed this concept, but an improved Hungarian algorithm is introduced in this paper, which gives the optimum result with reduced computational complexity. In addition to this, the proposed algorithm is most generalized one which solves the assignment problem for all possible number of machines and jobs, which is not addressed by other researchers. Furthermore, it ensures no machine or job is overloaded.

Keywords: Unbalanced Assignment, Improved Hungarian, Reduced computational complexity, No overload.

1. Introduction

Proper assignment of tasks or jobs is essential in so many fields like education, transportation, bio-medical, health care, mechanical, electrical, thermodynamics, sports, etc. [1] and in particular applications like selection of baseball team [2], Travelling Salesman Problem [3], efficient production in industrial aspects [4], tracking objects on camera which is a monitored transportation system [5], assigning accounts to accountants, tasks to bidders in auction based assignment, priority based channel allocation[6], prediction based channel assignment[7], research problems to researchers, products/area to distributers, patrolling areas to police men, workers to machines in industries and so on [8]. For example in [9], the authors reviewed various assignment problems like exam time table, course time table, new student allocation with all possible approaches. For assigning crops to paddocks Hungarian algorithm is used in [10].

When there are 'n' jobs and 'n' machines then assignment algorithm assigns the jobs to machines in one to one correspondence. Let C represents the cost matrix of size n*n and C_{ij} be the element of cost matrix, which represents cost corresponding to i^{th} machine assigned to j^{th} job. Here cost refers to time requirement or any other expenditure based on the application. The objective of the assignment problem is to find optimum assignment, which gives least total cost and maximum profit.

The assignment problem is basically of two types: Balanced and unbalanced assignment problems. In balanced assignment problem there are equal number of machines and jobs, whereas unequal number of jobs and machines are found in unbalanced problems.

To solve the assignment problem there are number of ways. Out of them linear programming, transportation and Hungarian algorithms are more popular ones. In linear programming approach, first the assignment problem is converted to optimization problem and usually solved with the help of simplex algorithm. Transportation algorithm is generally used when there are few sources and few destinations mentioned with variable supply and demand quantities respectively. To satisfy both sources and destinations transportation algorithms are used.

The classic solution to the assignment problem is Hungarian method which is originally proposed by H. W. Kuhn in 1955[11] and improved by J. Munkres in 1957[12]. In [1], they defined the assignment problem as maximum weighted bipartite matching problem. Hungarian method is the predominant, familiar, and generally used method to solve the assignment problems. In this, the optimum solution is obtained by making only one assignment in any row or column to ensure one to one assignment

[13]. Hungarian algorithm is mostly applicable in the areas where the total cost is to be minimized.

Hungarian algorithm gives optimum one to one assignment when there are equal number of machines and jobs. For unbalanced assignment problems, prior to solve it, dummy jobs/machines are to be added to convert the unbalanced in to a balanced problem. But the jobs which are assigned to dummy machines cannot be served in reality. So to avoid that problem duplication of required number of machines /jobs, that is multiple jobs (machines) are assigned to a single machine (job) is proposed in this paper. At most care is taken while selecting the duplicate machines/jobs to minimize the cost of final assignment. In addition to that the proposed algorithm ensures no overloading of particular machine/job.

The remaining part of the paper is covered as follows: Section2 deals with related works, mathematical formulation of the problem is described in section 3, proposed algorithm is explained in section 4, numerical example is illustrated in section 5, section 6 compares the proposed algorithm with existing algorithms and section 7 concludes the work.

2. Related Works

The unbalanced assignment problem is addressed by few researchers like Kumar, Betts and Rabbani. The author of [14] divided the problem in to sub problems. He considered rows as jobs and columns as machines. If number of jobs (n) is greater than number of machines (m) then he selected m jobs such that whose row-sum is minimum, as a sub problem. Like that it is repeated till the remaining number of rows (n2) is less than number of columns (m). Then he selected n2 columns such that they are having least column-sum, which is the last sub problem. After such division of main problem into sub problems, he solved all the sub problems with the help of Hungarian algorithm. The total cost of the assignment is sum of all the costs obtained in sub problems.

In [15], they first converted the given unbalanced problem in to balanced problem with the help of cloning the machines and adding dummy jobs as he considered n>m. Then he solved the problem by using Hungarian algorithm.

In [16] they didn't convert the given unbalanced problem (rectangular matrix) in to balanced problem. They considered 'm' machines as rows and 'n' jobs as columns. They considered the problem of n>m. They simply followed the Hungarian algorithm except for the cancellation of remaining zeros of that row after each assignment.

3. Mathematical Formulation

Consider there are 'm' machines (M_1, M_2, M_3, …M_m) and 'n' jobs (J_1, J_2, J_3, …J_n). The 'm' machines need to be assigned to 'n' jobs [17]. The objective is to minimize the total cost assignment (Z).

$$Z = \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} X_{ij}$$

Where C_{ij} is cost corresponding to i^{th} machine and j^{th} job assignment

X_{ij} represents assignment matrix, where $X_{i,j}=1$ if i^{th} machine is assigned to j^{th} job

$=0$ if i^{th} machine is not assigned to j^{th} job

Subject to the constraints

Case-1: If m=n

$$\sum_{j=1}^{n} X_{ij} = 1$$

$$\sum_{i=1}^{n} X_{ij} = 1$$

$$\forall i \in 1,2,\ldots\ldots\ldots m$$

$$\forall j \in 1,2,\ldots\ldots\ldots\ldots n$$

Case-2: If m<n

$$\sum_{j=1}^{n} X_{ij} = 1$$

$$\sum_{i=1}^{n} X_{ij} = 1$$

$$\forall i \in 1,2,\ldots\ldots\ldots m$$

$$\forall j \in 1,2,\ldots\ldots\ldots\ldots n$$

Case-3 If m>n

$$\sum_{j=1}^{n} X_{ij} = 1$$

$$\sum_{i=1}^{n} X_{ij} = 1$$

$$\forall i \in 1,2,\ldots\ldots\ldots m$$

$$\forall j \in 1,2,\ldots\ldots\ldots\ldots n$$

4. Proposed Algorithm

Consider there are 'n' jobs and 'm' machines.

Case-1: If m=n then it follows Hungarian algorithm for assignment since it is simple balanced assignment problem.

Case-2: If m<n then

a. Convert in to balanced problem with the help of the following logic. While (n>m)
If (n mod m == 0)
Duplicate all the machines 'k' times where k=floor (n/m)-1. Else
Duplicate K2 machines which gives least row-sum, where K2=(n mod m) End if
End while

b. Follow the Hungarian algorithm for assignment.
Case-3: If m>n then

(a) Convert in to balanced problem with the help of the following logic. While (m>n)
If (m mod n == 0)
Duplicate all the jobs 'k' times where k=floor (m/n)-1. Else
Duplicate K2 jobs which gives least column-sum, where K2= (m mod n) End if
End while

(b))Follow the Hungarian algorithm for assignment.

5. Numerical Illustration of the Proposed Method

Let us consider the example of five machines and eight jobs, which is taken for illustration by [14,15,16] is given in Table 1.

Table 1: Assignment Problem Description

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8
Machine1	300	250	180	320	270	190	220	260
Machine2	290	310	190	180	210	200	300	190
Machine3	280	290	300	190	190	220	230	260
Machine4	290	300	190	240	250	190	180	210
Machine5	210	200	180	170	160	140	160	180

Step-1: According to the proposed algorithm first it needs to be converted to square matrix that is balanced assignment problem. Here number of columns is more compared to number of rows by three. So it needs three extra rows to be appended and the extra rows are selected starting from least row-sum towards highest row-sum. The resultant square matrix is shown in Table 2.

Table 2: Resultant Square Matrix

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8
Machine1	300	250	180	320	270	190	220	260
Machine2	290	310	190	180	210	200	300	190
Machine3	280	290	300	190	190	220	230	260
Machine4	290	300	190	240	250	190	180	210
Machine5	210	200	180	170	160	140	160	180
Machine2	290	310	190	180	210	200	300	190
Machine4	290	300	190	240	250	190	180	210
Machine5	210	200	180	170	160	140	160	180

Step-2: Identify row minimum of each row and subtract it from each element of that row, resulting in Table 3.

Table 3: Resultant Matrix after Row Minimization

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8
Machine1	120	70	0	140	90	10	40	80
Machine2	110	130	10	0	30	20	120	10
Machine3	90	100	110	0	0	30	40	70
Machine4	110	120	10	60	70	10	0	30
Machine5	70	60	40	30	20	0	20	40
Machine2	110	130	10	0	30	20	120	10
Machine4	110	120	10	60	70	10	0	30
Machine5	70	60	40	30	20	0	20	40

Step-3: Identify column minimum of each column and subtract it from each element of that column, resulting in Table 4.

Table 4: Resultant Matrix after Column Minimization

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8
Machine1	50	10	0	140	90	10	40	70
Machine2	40	70	10	0	30	20	120	0
Machine3	20	40	110	0	0	30	40	60
Machine4	40	60	10	60	70	10	0	20
Machine5	0	0	40	30	20	0	20	30
Machine2	40	70	10	0	30	20	120	0
Machine4	40	60	10	60	70	10	0	20
Machine5	0	0	40	30	20	0	20	30

Step-4: Start assigning from row 1 to row 8 and from column 1 to column 8, which are having single zeros. Whenever assignment is made, strike off the remaining zeros of that column/row to avoid multiple assignments to same job/machine. After all such assignments if still multiple zeros are observed in rows/columns and still pending of some assignments then randomly break the tie by selecting a zero randomly for assignment out of multiple zeros and strike off remaining zeros of that row/column. The resultant assignment is shown in Table-5, where the bolded zeros indicate the assignments.

Table 5: Resultant Matrix after First Assignment

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8
Machine1	50	10	**0**	140	90	10	40	70
Machine2	40	70	10	**0**	30	20	120	0
Machine3	20	40	110	0	**0**	30	40	60
Machine4	40	60	10	60	70	10	**0**	20
Machine5	**0**	0	40	30	20	0	20	30
Machine2	40	70	10	0	30	20	120	**0**
Machine4	40	60	10	60	70	10	0	20
Machine5	0	**0**	40	30	20	0	20	30

Step-5: Here it is observed that only seven assignments are made instead of all eight assignments. So it is required to modify the cost matrix. Tick the unassigned row and tick the column corresponding to zeros of that row and tick the rows corresponding to assignments of ticked columns. Now draw imaginary lines for unticked rows and ticked columns to cover all zeros. Now identify the least element, which is not covered by imaginary lines and subtract it from each uncovered element and add to the elements which are covered twice. After these operations the resultant cost matrix is shown in Table 6

Table 6: Resultant Matrix after First Modification

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8
Machine1	50	10	**0**	140	90	10	50	70
Machine2	40	70	10	**0**	30	20	130	0
Machine3	20	40	110	0	**0**	30	50	60
Machine4	30	50	0	50	60	**0**	0	10
Machine5	**0**	0	40	30	20	0	30	30
Machine2	40	70	10	0	30	20	130	**0**
Machine4	30	50	0	50	60	0	**0**	20
Machine5	0	**0**	40	30	20	0	30	30

Step-6: Repeat Step-4 for the resultant table after the first modification, which is Table 6. It is observed that all eight assignments are done, which are shown in Table-6 itself in bold.

Step-7: Calculate the cost corresponding to the assignments made from the original cost matrix. So the cost of the assignment becomes 180+180+190+190+210+190+180+200=1520.

6. Comparison of the Proposed Algorithm with the Existing Ones

The cost of the proposed algorithm is less compared to the algorithm proposed by [14]. The comparative costs of four different algorithms including the proposed one are shown in Fig.1 for the problem shown in Table 1.

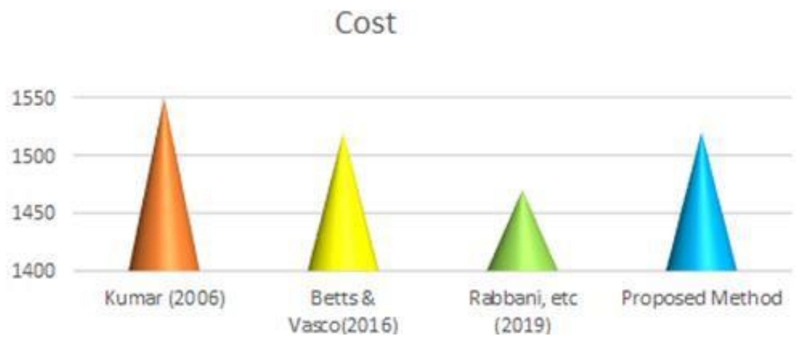

Fig. 1: Cost Comparison for four different Hungarian Algorithms for Table1 Problem

The cost of the proposed algorithm is same as the algorithm proposed by [15], but the computational complexity of the proposed algorithm is less compared to as their algorithm needs to solve a 10*10 matrix, where as our algorithm needs only 8*8 matrix to solve the same taken problem [16]. In general, to solve an n*n Hungarian assignment problem the computational complexity is O (n^4). So, the computational complexity for the algorithm proposed by [16] is an order of 10000, whereas it is just an order of 4096 only for the proposed algorithm as shown in Fig.2.

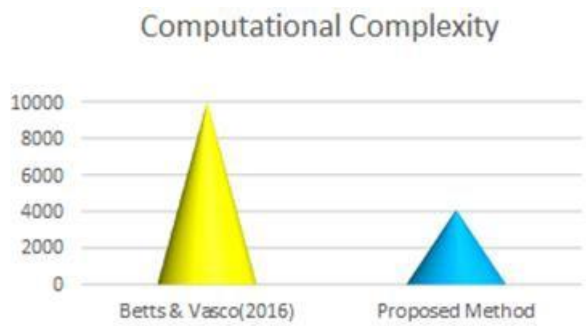

Fig.2: Computational Complexity Comparison

The cost of [16] is less than the proposed algorithm, but there are two problems with [16]. One of them is few machines got overloaded and this algorithm is not giving optimum solution for all problems. For the problem shown in Table- 7, the cost of their [14] (Rabbani 2019) algorithm is 75, whereas the cost of our algorithm, [14] and [15] is 74 as shown in Fig. 3.

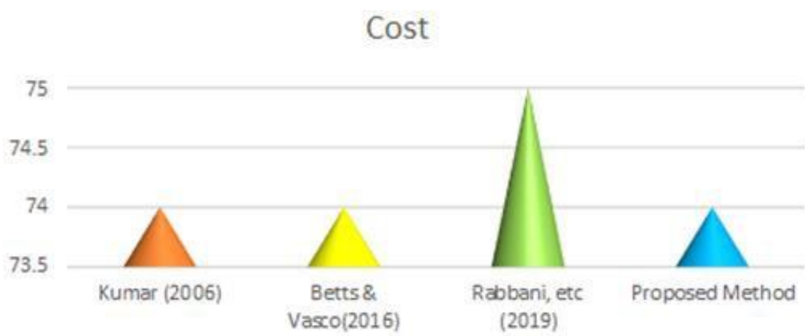

Fig.3: Cost Comparison for four different Hungarian Algorithms for Table 7 Problem

Table 7: Example Problem

	Job1	Job2	Job3	Job4	Job5
Machine1	28	10	15	30	12
Machine2	29	16	12	14	19
Machine3	32	8	25	20	14

The other problem with [16] is machine overloading. In the remaining works, the researchers have taken care of the amount of over loading on each machine. Due to this they [16] got less cost by overloading the machine, which is offering less cost. The amount of overloading for the previous three algorithms and proposed algorithm are shown in Fig.4 for the problem shown in Table 7.

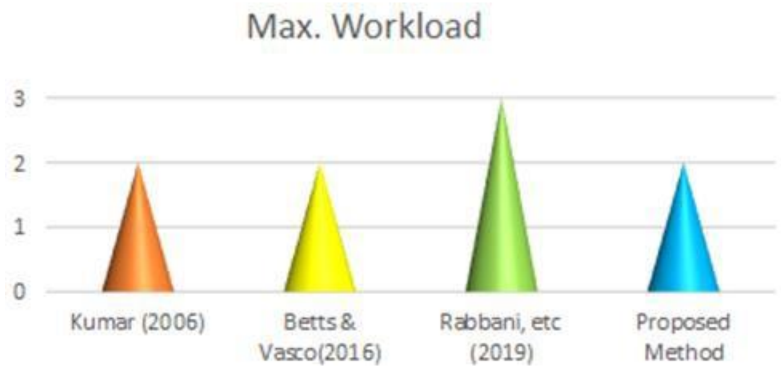

Fig.4: Maximum Workload Comparison among the four different Hungarian Algorithms.

In [18] they use heuristic algorithms PSO (Particle Swarm Optimization) and GA (Genetic algorithm). But these heuristic algorithms are preferred to solve complex problems, where the solution starts at random assignment and start moving towards optimum solution and sometimes it may not end with optimum solution. But the Hungarian is an analytic algorithm which is simple and assures optimum solution.

7. Conclusions and Future Scope

The proposed improved Hungarian algorithm gives optimum solution with less mathematical complexity for unbalanced assignment problems. Here all machines are being utilized and no machine is overloaded like [16] and no job is left unassigned. The proposed algorithm is giving optimum solution for any given problem.

The proposed algorithm can be applied for assigning sub carriers to secondary users in cognitive radio networks[19], time tabling problem in educational institutions [20], Placement of Staff in LIC using Fuzzy Assignment Problem [21], self-management aware autonomic resource management technique [22]to get optimum cost, etc.

References

1. Zeng, Y,Wu, X,Cao, J, (2014) Research and Implementation of Hungarian Method Based on the Structure Index Reduction for DAE Systems. Journal of Algorithms & Computational Technology 8:219-231.
2. Britz S.S, vonMaltitz M.J, (2010) Application of the Hungarian Algorithm in Baseball Team Selection and Assignment. University of the free state: 1-51.
3. Czopik, Janusz, (2019) An Application of the Hungarian Algorithm to Solve Travelling Salesman Problem. American Journal of Computational Mathematics 9: 61-67.
4. Kolinski, Adam, Koliński, Marek, (2013) The Use Of Hungarian Method In The Evaluation Of Production Efficiency: 116- 127.
5. Luetteke F, Zhang X, Franke J, (2012) Implementation of the Hungarian Method for object tracking on a camera monitored transportation system, ROBOTIK 2012; 7th German Conference on Robotics: 1-6.
6. K. S. Kumar, K. Annapurna and B. SeethaRamanjaneyulu, "Supporting real-time traffic in cognitive radio networks," 2015 International Conference on Signal Processing and Communication Engineering Systems, 2015, pp. 482-485, doi: 10.1109/SPACES.2015.7058201.
7. K.Annapurna, B.Seetha Ramanjaneyulu, C.Lakshmi Chaitanya, T.Hymavathi, "Spectrum Prediction in Cognitive Radio Networks using Neural Networks", International Journal of Control Theory and Applications, vol.10, issue.28, May 2017.
8. Kumar, P. Senthil, (2020) Developing a New Approach to Solve Solid Assignment Problems Under Intuitionistic Fuzzy Environment. International Journal of Fuzzy System Applications 9: 1-34.

9. Syakinah Faudzi, Syariza Abdul-Rahman, Rosshairy Abd Rahman,(2018) An Assignment Problem and its application in Education Domain: A Review and Potential Path, Advances in Operations Research. 1-19.
10. Lone, Mushtaq, Mir, S, Wani M, (2017) An Application of Assignment Problem in Agriculture Using R. Journal of Scientific Research and Reports 13: 1-5.
11. KUHN, H. W, (1955) The Hungarian method for the assignment problem. Naval Research Logistics Quarterly 2: 83–97. [12] MUNKRES.J, (1957) Algorithms for the assignment and transportation problems, Journal of the Society for Industrial and Applied Mathematics 5: 32-38.
12. Akpan N, P, Abraham U. P, (2016) A Critique of the Hungarian Method of Solving Assignment Problem to the Alternate Method of Assignment Problem by Mansi. International Journal of Sciences: Basic and Applied Research, 29: 43-56.
13. Kumar A, (2006) Modified method for solving the unbalanced assignment problems, Applied Mathematics and Computation. 176: 76–82.
14. Betts N, Vasko F.J ,(2016) Solving the unbalanced assignment problem: Simple is Better, Am. Journal of Operations Research. 6: 296–299.
15. QuazzafiRabbani, Aamir Khan, Abdul Quddoos ,(2019) Modified Hungarian method for unbalanced assignment problem with multiple jobs, Applied Mathematics and Computation. 361: 493-498.
16. Gillett Billy, E, (2000) Introduction to Operations Research – A Computer Oriented Algorithmic Approach, Tata Mc-GrawHill, NewDelhi
17. Pal,S., Khan, I., Maiti, M. K. (2018) A heuristic approach to solve multidimensional assignment problem,4th International Conference on Recent Advances in Information Technology (RAIT), Dhanbad: 1-7.
18. Annapurna K, Seetha Ramanjaneyulu B, (2019) QoS Maintenance in Cognitive Radio Networks with Priority-Supporting Novel Channel Allocation Method. International Journal of Business Data Communications and Networking: 1-16.
19. Mirhassani S, Habibi A, Farhang, Solution approaches to the course timetabling problem, Artificial Intelligence Review AIR. 39. 10.1007/s10462-011-9262-6. (011).
20. Trupti A Thakre , Onkar K Chaudhari ,Nita R Dhawade, (2018) Placement of Staff in LIC using Fuzzy Assignment Problem. International Journal of Mathematics Trends and Technology 53: 259-266.
21. Bhupesh kumar dewangan, Amit agarwal, Venkatadri m, Andashutosh pasricha, (2019) Design of self management aware automatic resource scheme in cloud, International journal of computer information systems and industrial management applications.11: 170-177.