Research Journal of Computer and Information Technology Sciences ISSN 2320 – 6527
Vol. **5(3),** 7-15, May **(2017)** *Res. J. Computer and IT Sci.*

Route optimization using Hungarian method combined with Dijkstra's method in home health care services

Monika Sharma[1*] and Rajesh Tiwari[2]
[1]Computer Technology and Applications, SSGI-SSTC, Junwani, Bhilai, CG, India
[2]Computer Science and Engineering, SSGI-SSTC, Junwani, Bhilai, CG, India
monikaofficialdocs@gmail.com

Available online at: www.isca.in
Received 10th April 2017, revised 16th May 2017, accepted 19th May 2017

Abstract

In this paper, we introduce a new approach for solving the route optimization problems and provide a solution for variant of this problem. The concept of proposed method is to combine the Dijkstra's method with Hungarian algorithm to find an optimal solution for a given problem. Presently the proposed approach is applied to a home health care system which deals with the providing medical care and emergency services to the patients. The method was explained with the help of an example and same can be implemented for the other applications also. The proposed method builds on the concept of Dijkstra's method and Hungarian method which is very simple, easy to understand and apply.

Keywords: Dijkstra's methods, Hungarian methods, Home health care, Route optimization.

Introduction

Home Health Care services (HHC) are becoming a gradually significant problem in the last few years, covering a great variety of decisions in Operational Research context[1]. Advantages of these services concern old age and/or people expressing different needs (such as medicines, medical care, intensive care, etc.), by offering them the opportunity to live in their homes with an equivalent medical follow-up to that given in Hospital[2].

Research on this field includes a fine coordination of human and material resources to provide an optimized planning that maximizes the quality of home health care while controlling generated costs.

In this paper, we propose to simplify the traditional vehicle routing problem to tackle new particular characteristics related to home health care[3]. The main goal is to optimized routes to be performed by an available vehicle (related to caregiver) and fit to visit a set of geographically distributed customers (related to the patients), who have preferences towards caregiver, and so that the activity is planned in the most effective way[4]. The originality of this study relies on the fact that several care services may be required to visit a particular set of customers either to supply some delicate medicines or giving some sort of insulin or to collect samples etc. within the least possible time.

Thus, the route between each node, firstly, should be optimized followed by the optimization of entire nodes. During the optimization it should be kept in mind that each node should be visited once and only once at a time and should return to its origin at the end.

In the home health care services center their problem is occurs in that way when their geographical manner there demand of services is generated. And service center has to provide services to that area where demand is generated.

Our main motto is to provide no. of services in sort time or as soon as possible. In over problem their special client who have emergency case their service must be provide first then proceeds further.

By shorting no. of service provider at minimum time we should have to short time first at only initial to next node after choosing these nodes we apply Dijkstra's method to choosing shortest path for only two nodes[5]. In these Dijkstra's method their initial and final node is fixed only path will be shorted by these methods and so on next node calculate their distance too next to next node or initial node then which path is shortest is chosen by the Dijkstra's method is fixed as path of these route.

After solving their node shortest path their big question is arise that the service provider has to provide no. of services of no. of people at minimum time then how can he/she provide services in short time.

For solving these problem their one more method is used Hungarian method these method is used to optimized no. of route and visited node by optimum path and short period of time so that the cost is proportionally decreases according to time of travelling.

Most important point is here during visiting time of providing services their time for waiting is fixed in each node. Means that the time taken to provide services at each node is fixed and in during service providing time if their one or extra demand is occurred by the customer that time these order be provided next time not that same time. In the visiting time their condition is that service provider provide service only once at a time.

Methodologies

Dijkstra's method: Edsger Wybe Dijkstra 11 May 1930 – 6 August 2002 was a Dutch computer scientist and an early developer in many research areas of computing science. He is describing the method of shorting shortest path of the graph. According to the Dijkstra's their vertices is define and edges is describing by following these techniques should find optimum vertices followed by choosing best solution for reaching initial node to end node. A method similar to Prim's algorithm is Dijkstra's method used as the optimization technique. In this, we generate a path for shortest route from the given source node already defined.

Two sets should be maintained, one contains vertices comprised of shortest path, other includes vertices not yet considered in shortest path. At each step of the algorithm, a vertex is to be found which is in the other set and has least distance from source. Assume the path is started is initial node and last visited node is final node.

Considering the distance of node is the distance of each node to next nearest node. Dijkstra's algorithm will define some distance values in between nodes and will try to improve them gradually: i. Allocate to every node a distance value: set it to zero for the initial node and to infinity for all other nodes. ii. Mark all nodes as unvisited and set initial node as current node. iii. For current node, consider all its unvisited nearby nodes and compute their distance from current node. iv. Assume all neighbors of the current node, mark them as visited node. A visited node will be checked once more by reverse its current node to initial node, optimum distance will be recorded and compared for the final distance between the nodes. v. If final node has been reached, then terminate the further search. Or else, set the unvisited node with the lowest distance (from the initial node to final node, all nodes in the graph are need not to be visited.

For example: Let assume there is a caregiver who have responsibility to deliver their medicine at Very short time, that time this caregiver has to deliver their medicine at very short time of period, so that he chose shortest path for delivery and hence it affected his travelling cost also by using Dijkstra's method.

In these graph their distance between different nodes are given and their initial node is A and final node is J where our medicine has to deliver in short time.

We have to deliver medicine in short time for that caregiver choose shortest path and major point is that caregiver never visited each node. He / she visit each path which is short in distance from initial node to final node.

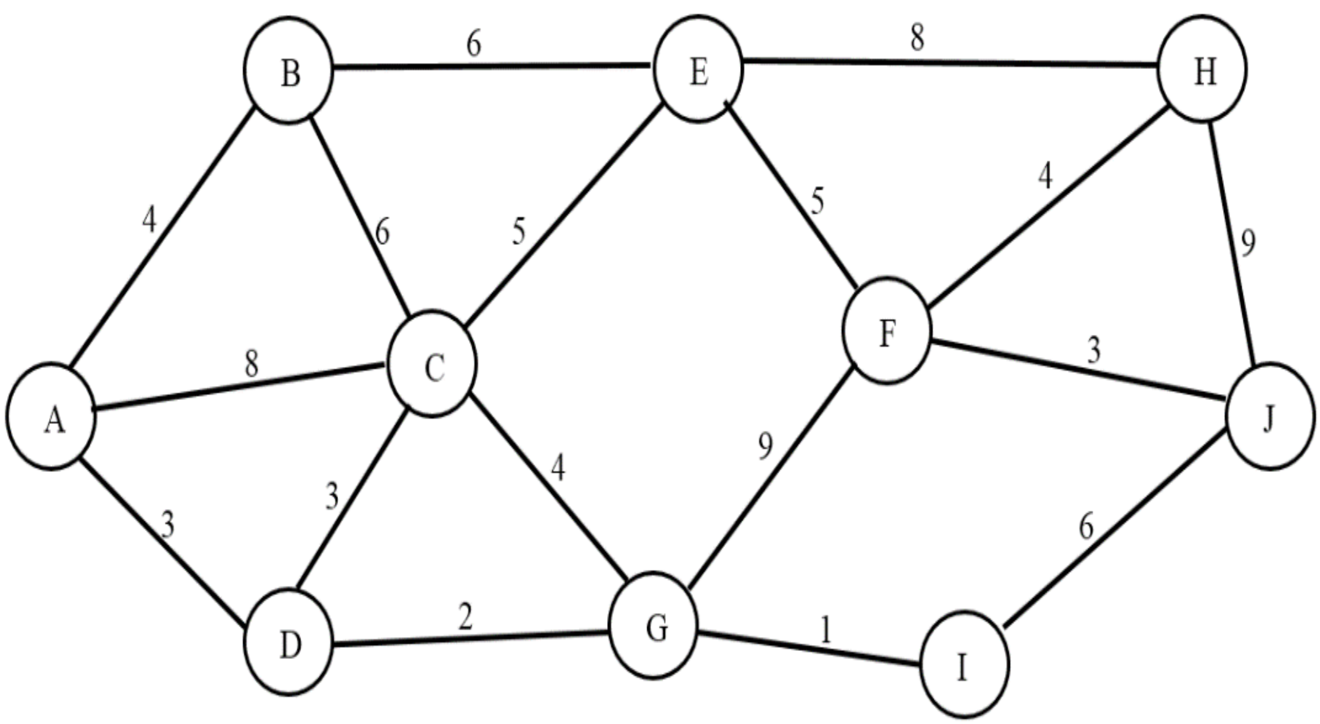
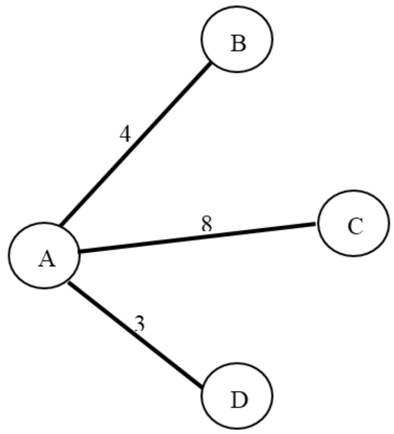
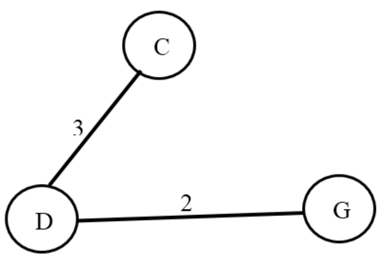


Figure-(a): Node A is connected to three other node B, C and D. those distance are A to B is 5, A to C is 8 and A to D is 3. After comparing these distance shortest one is chosen which is A to D is 3.



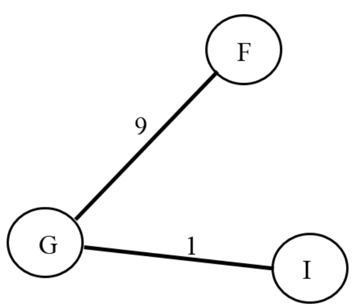
Path	Distance
a➔b	5
a➔c	8
a➔d	3

Figure-(a): Node D is connected to two other node C and G. The distance from D to C is 3 and D to G is 2. After comparing these distance shortest one is chosen which is D to G is 2



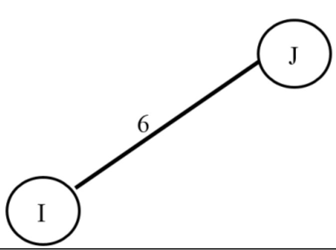
Path	Distance
d➔c	3
d➔g	2

Figure-(a): Node G is connected to two other node F and I. those distance is G to F is 9 and G to I is 1. After comparing these distance shortest one is chosen which is G to I is 1.



Path	Distance
g➔f	9
g➔i	1

Figure-(a): Node I is connected to node J. the distance between I to J is 6 there are only one path that it will be chosen.



Path	Distance
i➔j	6

Hungarian method: The Hungarian method is a very popular method to solve the assignment problem. Harold Kuhn in 1955 first devised and published this method and gave the name "Hungarian method". It was named so because the algorithm formed its basis on the earlier works of two Hungarian mathematicians: Jenő Egerváry adn Dénes Kőnig[6,7].

In 1957 James Munkres[8] reviewed the algorithm and found it more polynomial. From then onwards this method was also referred as Kuhn-Munkers algorithm. Mr. koning of Hungarian or the reduce matrix method or the floods technique is used for solving assignment problem since it is quite efficient and result in substantial time saving over other technique. The method consists of the following steps:

Prepare matrix: these steps will sort be require for n×n assignment problem, form m×n problem (where m is not equal to n), a dummy problem or dummy row as the case may be, is added to make a matrix square.

Reduce the matrix: i. Subtract the minimum element of each row of all the element of the rows. So there will be least one zero in each row. ii. Examine if there is a least one zero in each column, if not, iii. Subtract the minimum element of every column of all the element of the columns. So there will be least one zero in each column.

There steps reduce the matrix until zero, called zero opportunity cost are containing in each column Check whether the optimal assignment will be made in the optimal matrix or not for these: i. Examine row sequentially until row will until unmark zero is obtain. Make an assignment to this to their single zero by making box around it. cross all other zero in the same column as they won't be considered in any other assignment in that column process in that way until all zero have been examining. ii. Examine column sequentially until column will until unmark "zero" is obtain. Make a box around it and make it an assignment to this to their single zero. Cross out all remaining zero in the same row.

In case there's no row and no column containing single unmark zero, mark box any unmark zero whimsical any cut all alternative zero in its row and column. Proceed in same manner until there's no unmark zero left within the cost matrix. Repeat sub steps above until one in every of the subsequent two things occur: i. there is one assignment in every row and every column. In these case the optimum assignment are often created within the recent resolution, that's the recent possible resolution is an optimum solution. The minimum no. of line cutting all zero is n, the order of the matrix. There is a few row and a few columns without assignment. In these cases optimum assignments are often created within the current resolution. The lowest no. of line cutting all zeros have to be compelled to be locating during this case by following.

Find the minimum no. of line cutting all zero. This consists of the following sub steps: i. tick (✓) the row that does not have assignment. ii. tick (✓) the column that have zero in ticked column. iii. tick (✓) the row that have assignment in ticked column. iv. repeat sub steps above two till none of the row or column can be marked. v. All unticked row and unticked column draw straight line. This gives the minimum no. of line cutting all zeros. Its no. is equal to the order of the matrix, then it is an optimal solution, otherwise go to step 5.

Iterate toward then best solution. Examine the unticked components. Choose the bottom part and calculate it from all the uncovered components. Add this lowest part to each row part and calculate it to all or any the uncovered part. Add this lowest part to each part that lies as the intersection of two lines. Leave the remaining part of the matrix as such, this yield another basic possible solution.

Repeat step 3 though successive until the no. of line crossing all.

For example: In the factory survey is happening for machines, manager who have to come checking their machine performance. He has to visit each station and checking each machine there are distance between stations to main station are show in tabular form.

Here M is defining no. of machine and S is defining no. of stations.

	S1	S2	S3	S4	S5	S6	S7	S8
M1	67	2	90	50	22	61	47	17
M2	37	13	72	49	3	41	91	17
M3	21	94	46	39	40	4	14	5
M4	9	34	34	56	70	68	79	92
M5	82	38	4	17	99	94	98	72
M6	19	97	65	61	34	41	56	65
M7	27	23	96	83	35	22	7	14
M8	8	51	63	17	80	16	13	9

Solution of the problem is given below:
Step-1: Minimize each row: for minimization of row, subtracting least element of that row to every element of the row until there at least one zero.

	S1	S2	S3	S4	S5	S6	S7	S8
M1	65	0	88	48	20	59	45	15
M2	34	10	69	46	0	38	88	14
M3	17	90	42	35	36	0	10	1
M4	0	25	25	47	61	59	70	83
M5	78	34	0	13	95	90	94	68
M6	0	78	46	42	15	22	37	46
M7	20	16	89	76	28	15	0	7
M8	0	43	55	9	72	8	5	1

Step-2: Minimize each column: for minimization of column, subtracting least element of that column to every element of the column until there at least one zero.

	S1	S2	S3	S4	S5	S6	S7	S8
M1	65	0	88	39	20	59	45	14
M2	34	10	69	37	0	38	88	13
M3	17	90	42	26	36	0	10	0
M4	0	25	25	38	61	59	70	82
M5	78	34	0	4	95	90	94	67
M6	0	78	46	33	15	22	37	45
M7	20	16	89	67	28	15	0	6
M8	0	43	55	0	72	8	5	0

Step-3: draw straight line which is cutting only one zeros from each column and row also. If their no. of line is equal to no. of row then our solution is final solution. There no. of rows are 8 but there no. of line are 7. Otherwise follow step:4

	S1	S2	S3	S4	S5	S6	S7	S8
M1	65	0	88	39	20	59	45	14 ✓
M2	34	10	69	37	0	38	88	13 ✓
M3	17	90	42	26	36	0	10	0 ✓
M4	0	25	25	38	61	59	70	82 ✓
M5	78	34	0	4	95	90	94	67
M6	0	78	46	33	15	22	37	45 ✓
M7	20	16	89	67	28	15	0	6 ✓
M8	0	43	55	0	72	8	5	0

Step-4: Here no. of line or ticked mark is 7 and no. of rows is 8 which are not equal. So first of all view only that area where lines not covered, and finding the least value from those rows which is 15, subtracting that element from that row where line not covered. If there is any negative value than subtract that negative value and make it as zero.

	S1	S2	S3	S4	S5	S6	S7	S8
M1	80	0	88	39	20	59	45	14
M2	49	10	69	37	0	38	88	13
M3	32	90	42	26	36	0	10	0
M4	0	10	10	23	46	44	55	67
M5	93	34	0	4	95	90	94	67
M6	0	63	31	18	0	7	22	30
M7	35	16	89	67	28	15	0	6
M8	15	43	55	0	72	8	5	0

Step-5: After calculation of that matrix we will again find no. of zeros in rows and column. Follow step: 3.

	S1	S2	S3	S4	S5	S6	S7	S8
M1	80	0	88	39	20	59	45	14
M2	49	10	69	37	0	38	88	13
M3	32	90	42	26	36	0	10	0
M4	0	10	10	23	46	44	55	67
M5	93	34	0	4	95	90	94	67
M6	0	63	31	18	0	7	22	30
M7	35	16	89	67	28	15	0	6
M8	15	43	55	0	72	8	5	0

There again no. of lines are not equal to row so again follow step: 4.

	S1	S2	S3	S4	S5	S6	S7	S8
M1	87	0	88	39	27	59	45	14
M2	49	3	69	30	0	31	81	6
M3	39	90	42	26	43	0	10	0
M4	0	3	3	16	46	37	48	60
M5	100	34	0	4	102	90	94	67
M6	0	56	24	11	0	0	15	23
M7	42	16	89	67	35	15	0	6
M8	22	43	55	0	79	8	5	0

	S1	S2	S3	S4	S5	S6	S7	S8
M1	87	0	88	39	27	59	45	14
M2	49	3	69	30	0	31	81	6
M3	39	90	42	26	43	0	10	0
M4	0	3	3	16	46	37	48	60
M5	100	34	0	4	102	90	94	67
M6	0	56	24	11	0	0	15	23
M7	42	16	89	67	35	15	0	6
M8	22	43	55	0	79	8	5	0

This is over final solution their no. of rows and no. of line are equal. Taking single zero from row and column respectively.

	S1	S2	S3	S4	S5	S6	S7	S8
M1	87	0	88	39	27	59	45	14
M2	49	3	69	30	0	31	81	6
M3	39	90	42	26	43	0	10	0
M4	0	3	3	16	46	37	48	60
M5	100	34	0	4	102	90	94	67
M6	0	56	24	11	0	0	15	23
M7	42	16	89	67	35	15	0	6
M8	22	43	55	0	79	8	5	0

Following optimal solution in the original matrix:

	S1	S2	S3	S4	S5	S6	S7	S8
M1	67	2	90	50	22	61	47	17
M2	37	13	72	49	3	41	91	17
M3	21	94	46	39	40	4	14	5
M4	9	34	34	56	70	68	79	92
M5	82	38	4	17	99	94	98	72
M6	19	97	65	61	34	41	56	65
M7	27	23	96	83	35	22	7	14
M8	8	51	63	17	80	16	13	9

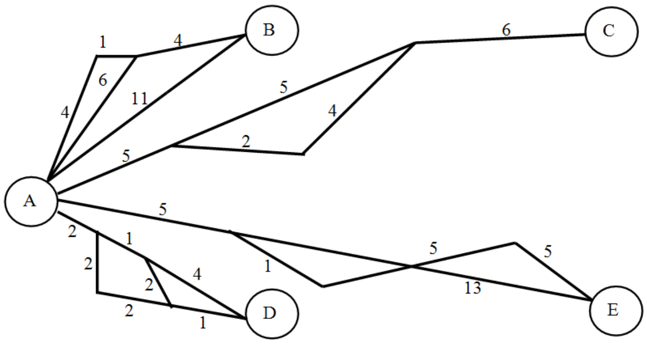
Optimal value is = 9+2+4+17+3+41+7+5= 88

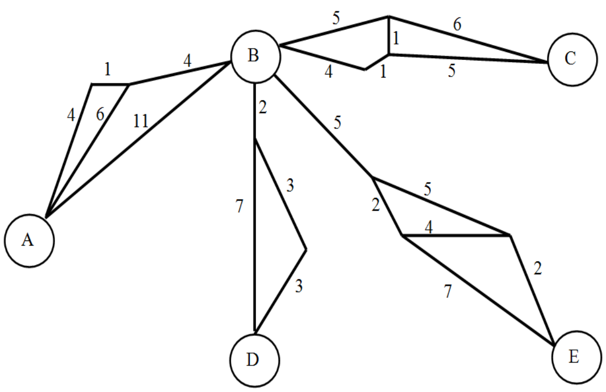
Route Optimization Using Hungarian Method Combined with Dijkstra's Method: In our problem we have to short the path and finding optimal solution to provide services at the patient home. As we consider there are no. of patient who need medicine and over responsibility to provide medicine in each patient in appropriate time.

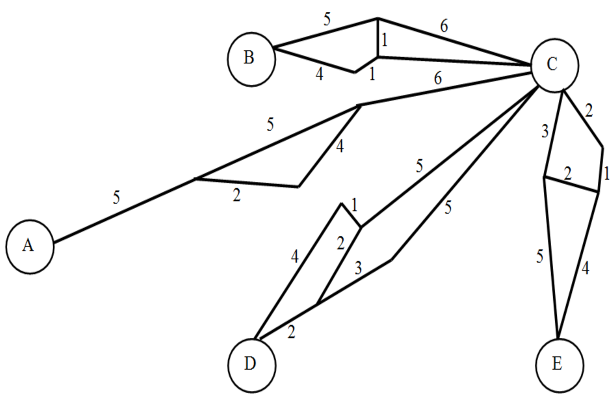
For providing medicine we have to short the path of patient home so that the caregivers reach patient home in short time but in the service providing time caregivers have to deliver the medicine at short time but in minimum cost is taken by caregivers so that patient need to short the path at each node or patient home also.

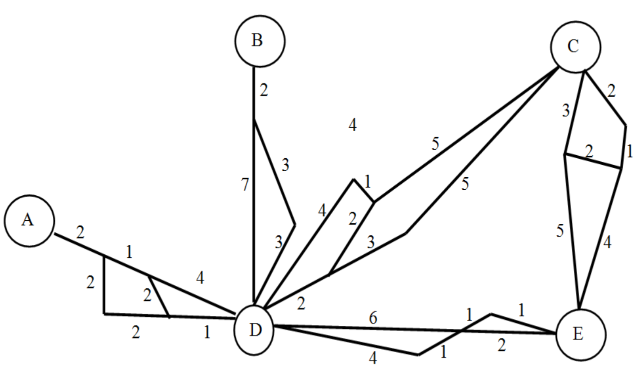
So that we have to take distance from service center to patient home distance and short the path to travel the distance
So that sorting path of each node we have to apply Dijkstra's method in each patient home to service center.

After that the shorting by applying Hungarian method.

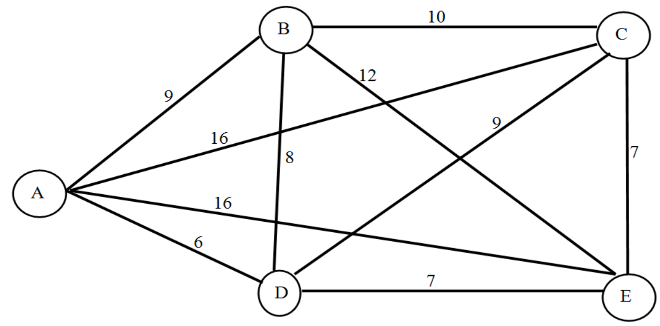








After applying Dijkstra's method to above problem it was found that the optimal route connecting different nodes are given as below:



By solving using Hungarian method the optimal solution is obtained as follows:

From/to	A	B	C	D	E
A	∞	9	16	6	16
B	9	∞	10	8	12
C	16	10	∞	9	7
D	6	8	9	∞	7
E	16	12	7	7	∞

First of all by applying Hungarian method the problem is converted into matrix form:

∞	9	16	6	16
9	∞	10	8	12
16	10	∞	9	7
6	8	9	∞	7
16	12	7	7	∞

Step-1: Subtracting row minima.

∞	3	10	0	10
1	∞	2	0	4
9	3	∞	2	0
0	2	3	∞	1
9	5	0	0	∞

Subtracting column minima.

∞	0	10	0	10
1	∞	2	0	4
9	0	∞	2	0
0	-1	3	∞	1
9	2	0	0	∞

Here column 2 having negative value so for removing negative value apply =1 in 4th row.

∞	0	10	0	10
1	∞	2	0	4
9	0	∞	2	0
1	0	4	∞	2
9	2	0	0	∞

The optimal assignment: Because there are 4 lines required, the zeros cover an optimal assignment:

∞	0	10	0	10
1	∞	2	0	4
9	0	∞	2	0
1	0	4	∞	2
9	2	0	0	∞

∞	9	16	6	16
9	∞	10	8	12
16	10	∞	9	7
6	8	9	∞	7
16	12	7	7	∞

In these stage the optimal solution is in form of cyclic because A➔D➔B➔A

The optimal assignment is cyclic but is does not cover each note to overcome these problem we consider the lowest entry 2 (B,C).

Now delete Brow and C column then consider zero assignment in remaining matrix

The resulting matrix is:

∞	0	10	0	10
1	∞	2	0	4
9	0	∞	2	0
1	0	4	∞	2
9	2	0	0	∞

∞	9	16	6	16
9	∞	10	8	12
16	10	∞	9	7
6	8	9	∞	7
16	12	7	7	∞

Here final solution is A==> B ==>C ==> E ==> D==>A
And cost of the matrix is - 9+10+7+6+7=39.

Conclusion

In this paper, we have developed the Hungarian method combined with the Dijkstra method for solving the problem route optimization. A numerical example has been solved using our proposed method. It was found that the proposed concept is helpful in solving present as well as future real-life problems in the area of route optimization. The method has wide area of application in the field of tour and travels, school buses, milk distribution, medical care services etc.

References

1. Cheng E. and Rich J.L. (1998). A home health care routing and scheduling problem. *Houston, Texas,* 24.
2. Mondal R.N., Hossain M.R. and Saha S.K. (2013). An Approach for Solving Traveling Salesman Problem. *International Journal of Applied Operational Research-An Open Access Journal*, 3(2), 16-26.
3. Liu R., Xie X., Augusto V. and Rodriguez C. (2013). Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care. *European Journal of Operational Research*, 230(3), 475-486.
4. Haddadene S.R.A., Labadie N. and Prodhon C. (2016). A GRASP× ILS for the vehicle routing problem with time windows, synchronization and precedence constraints. *Expert Systems with Applications*, 66, 274-294.
5. Skiena S. (1990). Dijkstra's algorithm. *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica, Reading, MA: Addison-Wesley*, 225-227.
6. Kuhn Harold W. (1955). The Hungarian Method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2), 83-97.
7. Kuhn Harold W. (1956). Variants of the Hungarian method for assignment problems, *Naval Research Logistics Quarterly*, 3(4), 253-258.
8. Munkres J.L. (1957). Algorithms for the Assignment and Transportation Problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1), 32-38.