Reflex Matrix-Minimum Sum Algorithm –Shortest Distance
(Application to Search for Minimum Spanning Tree and ItsUniqueness)

PRADEEP. J. JHA [1] and SWETA SHAH* [2]

Visiting Faculty, Department of Mathematics [1]
St. Xavier's Institute, Ahmedabad, Gujarat, India

Assistant Professor, Department of Mathematics[2]
Parul University, Vadodara, Gujarat, India.

Key words: Shortest Distance, Minimum Spanning Tree, Reflex Matrix

Abbreviations: M.S.T. (Minimum Spanning Tree), S.D. (Shortest Distance)

2010 Mathematics subject classification: 05C05 Trees, 05C50 Graphs and Linear Algebra

Abstract

Content of this unit depict alternative approach in the direction of finding shortest path in a given network wherein edges are assigned figures that indicate distance between the two vertices. Notion of Reflex matrix introduced in this unit help process technically speedy and exhaustively. As an application, minimum spanning tree in the given network is derived which need application of some different algorithms otherwise. Theorems establish soundness of the basics of work.

1. Introduction

It is known that the topic to searching for the routes indicating shortest distance between any two points in the given network diagram stands on its merits and so it is applicable in many real-life situation areas. There are a few algorithms available to resolve this problem technically and we have introduced a new one-- concept of 'Reflex Matrix'. This matrix helps find shortest distance between any two vertices in the given network and hence the shortest path. Application to this logic in different areas can simplify many real situation problems; we deal those in the second part of this paper.

Assumptions: There are certain assumptions to justify proceedings of this model.

1 There are no self-loops on any vertex of the network diagram.

2 There are no parallel edges or routes joining two distinct vertices.

1.1 Procedure:We now introduce algorithmic routine to arrive at the goal.

Here an attempt has been made to simplify the underlying logic by introducing an upper diagonal matrix having entries indicating distance between two vertices in the given network of 'n' vertices. Let i and j be any two vertices in a network diagram (i< j).

Let d(i, j) denote the distance between the vertices i and j [4].

We have

i. $d(i, j) \geq 0$
ii. d (i,j) = 0 if and only if i = j**(1)**
iii. d (i, j) = d(j, i)
iv. [Triangular inequality may hold]

1.2 Network and upper diagonal Matrix:

We consider a network diagram and find shortest distance between any two vertices and then find minimum spanning tree as an extension of shortest route.

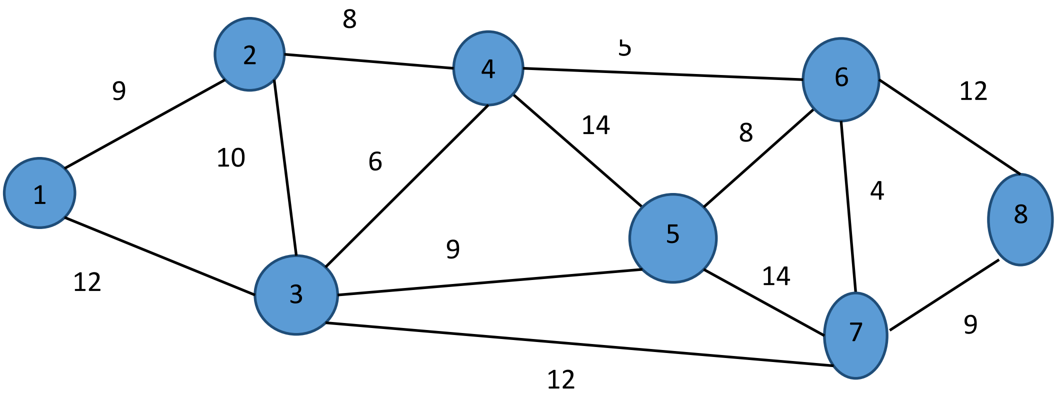

Figure 1: Network Diagram

Step-1: Analysis of the given network:

There are **8** vertices and **14** edges which are labelled with figure showing distance between them. We have n = no. of vertices =8 and no. of edges = E =14;

G = (n, E)s = (8 ,14)

We construct adjacency matrix = A; an upper diagonal one with all leading diagonal entries equal to zero [2].

Step-2 Designing Distance Matrix:

Design the distance matrix D whose entries are the distances between the vertices i and j;where iand j are integer values showing the vertices as given in the network. We follow notation given by **(1).** Blank entries show that there is **no** direct route between the two vertices.

1 Set d(i, i) = 0 entries on leading diagonal.

2 Set d(i, j) = d as the given distance from network diagram where i< j. [upper diagonal-Bold]

3 Set d(j, i) = d* where i> j and by **(1)** we have d = d* [lower diagonal - reflex]

Entries d and d* are reflexes of each other. Matrix D is called the reflex matrix [4].

Distance Matrix d (i, j)

Vertices1 2 3 4 5 6 7 8

1 0 **9** **12** **0** **0** **0** **0** **0**

2 9* 0 **10** **8** **0 0** **0** **0**

3 12*10* 0 **6** **9** **0** **12** 0

D = 408* 6* 0 **14** **5** **0** **0**

5 009*14*0**8** **14** 0

6 000 5* 8* 0 **4** **12**

7 00 12* 0 14*4* 0**9**

8 00000 12* 9*0

[The entry d12 = 9 indicates that as given in the network the direct distance between the vertices 1 and 2 is 9 units. As d12 = d21; we write d21= 9 as a reflex entry to the entry d12 and so on.]

Step 3Completion of Lower Diagonal Entries:

In this step we fill entries d(i, j) where i< j. [Entries with ‘0’ figure indicates that there is no direct path.]

In order to fill the remaining entries of the lower diagonal matrix we note the following points.

1 These entries are other than marked with * entries.

2 Begin with (i, j), with i> j and not marked with *

3 For a fixed i, we can consider j = 1, 2, 3 with j <i

4 For the entry (i, 1), i = 1, 2,3 -----, look the entries above the cell d(i, i)**[d(i, i) = 0].**

Now add each oneof such entry to the corresponding entries like d(i-1, 1), d(i-2, 1) etc.

find the minimum of all such resultsand fill the cell d (i, 1) with minimum value.

[We clarify some of the steps.

(a1): Say we want to fill **d(4, 1)**; look for all entries above the cell d (4, 4). The entries above it are d (3, 4) = 6 and d(2, 4) =8. Now add these entries tocorresponding entries d(3, 1) = 12 and d(2, 1) = 9. Now search for minimum and assign that minimum value to d(4,1)

d (3, 4) + d(3, 1) = 6 + 12 = **18**and d(2, 4) + d(2, 1) = 8 + 9 = **17**

minimum (18, 17) = 17 and this will allow to put d(4, 1) = 17.

(a2): Say we want to fill the cell d(5, 2)

Go to d (4, 5) = 14 and d(3, 5) = 9

Also look for d(4, 2) = 8 and d(3, 2) =10.

Therefore, d (5, 2) is the minimum of the sum d(4,5) + d (4,2) = 14 + 8 = 22 and that of d (3, 5) + d(3, 2) = 9 + 10 = 19. Minimum of d (22, 19) = 19 and so we have d(5, 2) = 19.

(a3): We find d (6,2). We have d(6,2) = minimum of d(5,2) + d(5,6) = 19 + 8 =27 and d (4,2) + d(4, 6) = 8 + 5 =13 which is 13. So we have d(6,2) =1

Following this pattern carefully, we fill all the entries of the lower diagonal matrix. This, at the end makes the matrix ready forfinding the shortest distance from the origin and terminusvertex and shortest distance between any two vertices in the network diagram [1].

We preparethis matrix—called destination matrix D1.

Destination Matrix D1 =**d (i, j)**

Vertices 1 2 3 4 5 6 7 8

1 0 **9** **12<** **0** **0** **0** **0** **0**

2 9* 0 **10** **8 <** **0** **0** **0** **0**

3 12* 10* 0 **6** **9** **0** **12<**0

D1 = 4 17 8* 6* 0 **14** **5<** **0** **0**

5 21 19 9* 14* 0 **8** **14** 0

6 22 13 11 5* 8* 0 **4** **12**

7 24 17 12* 9 14* 4* 0 **9<**

8 33 25 21 1720 12* 9* 0

Note:

There are some entries in the upper diagonal matrix which are responsible for selection of the shortest distance. An entry in j^{th} columnin the upper triangular region is highlighted with ‘<‘ sign which helps give the shortest distance the vertices j and 1= the entry at (j, 1) coordinate. E.G. The entry (4, 6) is 5 marked as ‘5 < ‘as it found useful to write d(i,1) for some i = 1,2.3…

d(6,1) = minimum of d(5,1) + d(5,6) = 21 + 8 = 29 and d(4,1) + d(4,6) = 17 + **5<**= 22

Etc.

1.3 Applications:

At this point the destination matrix D1 is ready for two applications.

(1) Shortest path between any two points in the network and (2) Minimum Spanning Tree

(A1) Shortest Path:

Entries in the lower triangular region are very important for deriving shortest distance

between

(a) Source and terminus, (b) any two vertices, and (c) Minimum Spanning Tree.(M.S.T.)

Step 4: Here we find solution of the abovementioned points.

(a) Shortest distance between source—vertex 1 and terminus—the last vertex –vertex 8 in this case.

Last vertex – 8, look for ‘< ‘in the last column.There is an entry at the point (7,8) which is 9<.

This means that the vertex 7 is on the shortest path.

Now, look in the 7^{th} column; find the entry with ‘<’.The entry in coordinate (3, 7) = 12<

This means that the vertex 3 is on the shortest path.

Now look in the 3^{rd}column searching for ‘<’ sign.It is located in the position (1,3) which appears as 12<

This means that vertex ‘3‘is on the shortest path. In the same way we find the source vertex 1 to be on the path [5].

This makes the entire path **1- 3 – 7–8 of 12 + 12 + 9**= 33 units of length.The shortest distance being the entry oncoordinate (8, 1) in the matrix which in this case is 33 units.

(A2) The shortest distance between any two vertices i and j,

As by (1), d(i, j) for i< j, is same as d (j, i); we find the value of the coordinate (j, i) from the matrix D1.

E.g., d (3, 6) is same as d(6, 3) which is equal to11 units found from the matrix D1.

1.4 Theorem1:Shortest path from origin to sink vertex, in general, may not be unique.

Proof: Let us consider a shortest path from origin vertex ‘1’ to the sink vertex ‘n’ in a network diagram. Say, there are k number of vertices involved on this path where $2 < k < n - 2$.

If $k = n - 3$then uniqueness is naturally established.

If $k < n - 4$ then it means that at least one vertex, say $'p'$, which does not participate in the shortest path.

Then the shortest path is joining the vertices $1, 2, \dots \dots p - 1, p + 1,\ n\ of\ n - 2$edges.

If the sum of distances $d\ (p - 1, p) + d(p, p + 1) = d(p - 1, p + 1)$

then we have two different paths from the vertex 1 to vertex ‘n’.

Both following the above criteria are shortest paths in the network. This proves non- uniqueness of the shortest path in the network diagram.

Note: *An example supporting the theme of the theorem has been given in the addendaa-1

(A3) Application: Minimum Spanning Tree (M.S.T)

It is known that a 'tree' is an acyclic connected graph.

A network diagram on N vertices and E number of directed edges, denoted as G (N, E), with edges depicting absolutedistance between any two vertices; say i and j,denoted as d(i, j) is given.

A minimum spanning tree (M.S.T.) is the connected acyclic graph with minimum sum of all thedistances on those edges lying on path. To achieve this, we have to search for only those vertices which are not on the shortest Path.

It remains to add those vertices which are not on the shortest path. A vertex which is not on the given path must be connected with any of the previous vertex which has minimum distance.

We join these two vertices provided it does not form a cycle and continue the process till all vertices are included. This is the minimum spanning tree[5]. We clarify this in our example.

In our network, as discussed above, the path 1- 3- 7- 8 is the shortest path. We need to add the remaining vertices 2, 4, 5, and 6 in a way that introduction of new edges does not form aloop and the total distance remains minimum. This is M.S.T.

As 1 3 7 8 is the shortest path we add vertices 2,4,5 and 6.

2 4 5 6 can be written in the order.

Steps towards M.S.T.-

1) Vertex 2is adjacent to the vertices 1, 3, and 4.

Edge:	1-2	2-3	2-4
Distance:	9	10	8

Minimum is on the edge2- 4 which is 8 units and introduction of the edge 2- 4 does not form a loop. We join the vertices 2-4

2) Vertex 4 is adjacent to the vertices 2, 3, 5, and 6.

Edge:	2-4	3-4	4-5	4-6
Distance:	8	6	14	5

The edge 2-4 is already used in the process. We include the edge 4-6 with minimum distance 5 units. We are at this junction wherein we need to connect the vertices 5 and 6 with main junction.

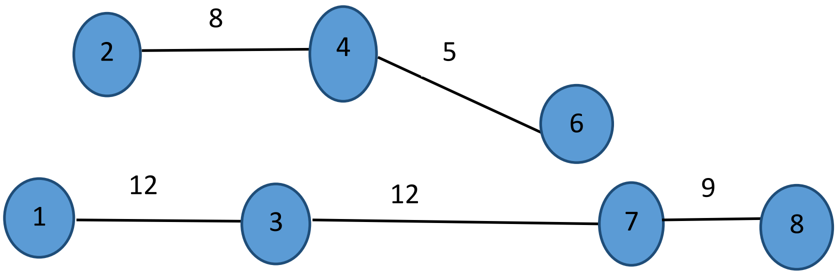

Figure 2: Minimum Spanning Tree

3)Vertex 5 is adjacent to the vertices 3, 4, 6, and 7.

Edge: 3-5 4-55-6 5-7

Distance: 9 148 14

We select the edge 5-6 with minimum distance = 8units. This makes it acyclic and connected too.

4) Finally, we are left with the vertex 6. For that we have

Edge: 4-6 5-6 6-7 6-8

Distance: 5 84 12

We have already used the edge 5-6 in the above case 3). We are left with the minimum choice of the edge 6-7 with minimum distance 4 units.

As a result of this modifications, we are atminimum spanning Tree—An acyclic connected graph with minimum weightbearing.

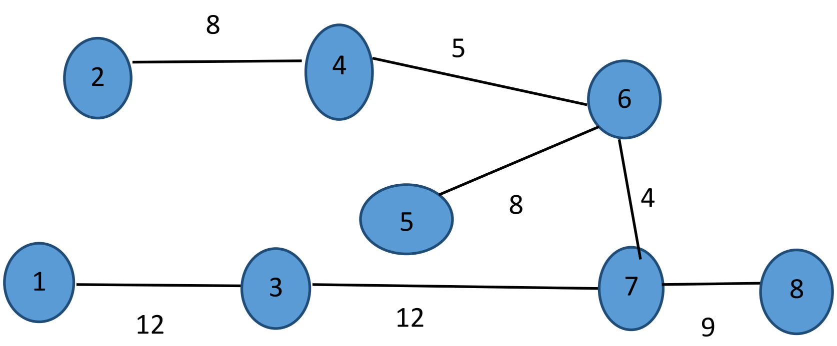

Figure 3: Minimum Spanning Tree (Total minimum weights =58 units)

In this way it makes easier to extend the basic notion of shortest distance to minimum spanning Tree.

Theorem: 2Minimum Spanning Tree (M.S.T.) is unique.

Proof: Let us consider a network diagram on ‘n’ distinct vertices.

Case-1: If the shortest path on this network comprises of (n-1) edges on this ‘n’ vertices then itself is a minimum spanning tree.

Case -2: If the number of vertices involved on the shortest path are less than ‘n’ then it cannot be a M.S.T. It is known that minimum spanning tree on all ‘n’ vertices is a connected andacyclic tree with minimumweight bearing edges

As discussed in thetheorem 1, if a particular vertex, say ‘P’ is not on the shortest path then we find a vertex adjacent to the vertex ‘P’ with minimum weight and join them by an edge. This makesminimum spanning tree.

This establishes the fact that minimum spanning tree is unique.

Conclusion

Reflex matrix technique discussed above develops an important extension that handles shortest path in a network and its non-uniqueness. A M.S.T and its uniqueness theoremis established as an immediate application.In addition to this there are further scopes ofcoordination between some important units of graph theory and matrix algebra which can prove its importance in real life situation.

References

[1] Jha,Pradeep.J., Operation Research, Mc Graw Hill Education(India) Pvt. Ltd. (2014) ISBN: 81-1-25-902673-7

[2] Clark, John and Holton, D. A, *A First Look at Graph Theory,* Allied Publishers Ltd, (1991) ISBN: 81-7023-463-8

[3] Angela, Mestre, ‘An Algebraic Representation of Graphs and Application to graph Enumeration’, International journal of Combinatorics, Vol. 2013,Article ID 247613

[4] Hazra, A.K., *‘Matrix Algebra, Calculus and Generalized Inverses’,* Part-1,Viva Books Pvt. Ltd, First Edition. (2009) ISBN:978-81-309-0952-3

[5] Singh, Lovejit, ‘Concept of Graph Theory and its Applications’, Journal o Engineering Technology and innovative Research, 6(5) (2019),120-123