ISSN : 2350-0743

www.ijramr.com

International Journal of Recent Advances in Multidisciplinary Research
Vol. 05, Issue 02, pp.3623-3625, February, 2018

RESEARCH ARTICLE

SIMULATION OF LINK-STATE ROUTING PROTOCOL

***Saikrishna, P.V., Rishidar, Nirudh, Akhil and Manikandan, K.**

Vellore Institute of Technology, Vellore (India)

ARTICLE INFO

Article History:
Received 16th November, 2017
Received in revised form
20th December, 2017
Accepted 22nd January, 2018
Published online 28th February, 2018

Keywords:

CSE - Computer Networks & Communication,
Dijkstras Algorithm.

ABSTRACT

In this paper, we give a subjective and quantitative study of The Simulation of Link-State Routing Protocol which is a part of the project under course CSE - Computer Networks & Communication. The program accepts the network topology details in terms of the cost of links, and provides the shortest path tree for the network. It uses Dijkstras Algorithm to construct its routing table.

INTRODUCTION

Link State is a type of protocol of routing, in computer networks and communication, it is used in switching of packet networks. In the network that uses link state, every node will perform link - state; they construct a map of the connection between nodes in network they also show the nodes that are connecting to other nodes in the network map by showing which nodes connected with which other nodes. After that every node can be separately calculate the shortest path from it to each and other nodes in network map.

Dijkstra's algorithm is one of the famous data structure algorithm best ever known for its single source which is from one node best shortest path Algorithm. It also calculates the low economic-cost path from the source node that is the start state to each and every other nodes in the network map. Dijkstra's algorithm is one of the repetitive and recursive algorithm and that has the special charecteristic that after the certain nth repetation of the process in the algorithm, the low and economic cost paths are also called as n-destination nodes or final nodes, and out of all of themlow and economic paths to all d estination nodes or final nodes, these n-paths would have n lowest paths.

Proposed Model

Link State Protocol

In general the link-state routing protocol means that each and every other node by which the network map is created, it would form in the form of graph in the network, this implies and portrait's which all nodes are connected to one another. Each and every node are separately calcuclates the best form of the path in logical way from itself to every other possible node destination in the network map or graph.the addition of every best paths in the tree will create the shortest path tree, it can be made as a routing table later. The Link state routingis made by thinking that each and every node in the network map has some perspective information: it should atleast know the cost from each state to other neighbouring links. It can also be said as , " The whole topology can be compiled from the partial knowledge of each node". In this way, it allows the topology to be dynamic. In link state routing,there arefour different steps that are essential to make sure that each and every node has a routing table showing the most economic node to every other node

- The node states of the connecting links created by each node, it is also known as link state protocol as well LSP.
- Dis persing of LSP data to all the remaining router, called flooding.
- Form a path tree which take minimum distance from each node.
- Calculate the table depending on the minimum path tree.

Dijkstras Algorithm

Dijkstra's algorithm is one of the famous data structure algorithm best ever known for its single source which is from one node best shortest path Algorithm. It also calculates the low economic-cost path from the source node that is the start state to each and every other nodes in the network map.

Corresponding author:* **P.V. Saikrishna,
Vellore Institute of Technology, Vellore (India).

- From table of n routers we will have n different shortest paths that form tree, each and every node would become the root node in its minimum shortest path tree.
- Make the minimum shortest path distance for each and every root node neighbours to the minimum cost between the neighbour and the node
- We continue to begin the search such that the every nodes in the way choose one with the low economic shortest distance and join to the path.
- Now use this operation: 55 = 555555 (55 , 55 + 55,5), we repeat above step until unless each and every node is added to the shortest path.
- While having the minimun shortest path tree we can create the routing table, we use 0 to represent the distance from root to root i.e. for same router say R1 → R1 has cost = 0. On the other hand it has positive cost values for two different nodes.

Ideas and Algorithms

The Dijkstras Algorithm to find the shortest path to from a source router to a destination router is implemented in Python for this project. The program follows below steps:

- The program first asks for a network topology file. It validates the data and store in matrix format.
- The next step is to create the connection table. The program takes the source router as input, and performs the Dijkstra Algorithm on it as explained in previous section. At every step, it keeps track of two type of nodes :
  - The interface used to go to next router. (For connection table.)
  - The parent node of last added node. (To create the final path.)
- Once both connection table and parent table are ready, the shortest path can be found from given source to destination by one of the two ways :

Starting from the source node, follow the interface from the connection table to reach to the destination. Starting from the destination node, follow the parent node from the parent table to reach to the source, and provide the reverse path. In both the cases, the path and total cost is found and returned to the user. If there is no path from given source and destination, the program returns with such message.

Functionalities

- Create a Network Topology
- Build a Connection Table
- Shortest Path to Destination Router
- Modify a topology
- Exit

Advantages

- Smaller routing tables.
- Faster convergence.

Disadvantages

- LSP require higher processing power when compared Distance Vector Protocols.
- Link State Routing Protocol require more memory.
- Initial flooding may degrade the performance of the network.

TEST RESULTS

The Link – State Routing Protocol implemented in this project using Dijkstra's algorithm has been tested under various cases and is found to be successful. This project has been tested for various numbers of routers like 5 routers, 10 routers, 12 routers, 15 routers, 20 routers etc. The successful and correct results were obtained during testing and the results are tabulated as below.

INSTANCE 1:

Number of routers = 5

Input File:

5_routers.txtOriginal Matrix :

```
 0  2  5  1 -1
 2  0  8  7  9
 5  8  0 -1  4
 1  7 -1  0  2
-1  9  4  2  0
```

Routing or Connection Table:

Shortest Path from Rooter: [1] to [5] is: 1 – 4 - 5
The total cost is 3

Shortest Path from Rooter: [3] to [1] is: 3 - 1
The total cost is 5

Shortest Path from Rooter: [5] to [4] is: 5 - 4
The total cost is 2

INSTANCE 2:

Number of routers = 8

Input File:

8_routers.txtOriginal Matrix :

0	43	42	44	-1	27	31	32
43	0	42	40	38	28	30	34
42	42	0	43	36	29	29	-1
44	40	43	0	36	30	28	36
-1	38	36	36	0	31	-1	38
27	28	29	30	31	0	27	40
31	30	29	28	-1	27	0	-1
32	34	-1	36	38	40	-1	0

Routing or Connection Table:

	N1	N2	N3	N4	N5	N6	N7	N8
N1		2	3	4	6	6	7	8
N2	1		3	4	5	6	7	8
N3	1	2		4	5	6	7	8
N4	1	2	3		5	6	7	8
N5	6	2	3	4		6	6	8
N6	1	2	3	4	5		7	8
N7	1	2	3	4	6	6		1
N8	1	2	9	4	5	6	1	

Shortest Path from Rooter: [2] to [6] is: 2 - 6
The total cost is 28
Shortest Path from Rooter: [4] to [8] is: 4 - 8

The total cost is 36

Shortest Path from Rooter: [8] to [1] is: 8 - 1
The total cost is 32

INSTANCE 3:

Number of routers = 10

Input File:

10_routers.txtOriginal Matrix :

0	2	-1	-1	-1	-1	-1	-1	1	5
2	0	2	-1	-1	-1	-1	-1	-1	-1
-1	2	0	4	-1	-1	-1	-1	-1	-1
-1	-1	4	0	2	-1	-1	-1	4	2
-1	-1	-1	2	0	-1	4	-1	-1	-1
-1	-1	-1	-1	-1	0	2	-1	-1	5
-1	-1	-1	-1	4	2	0	6	-1	-1
-1	-1	-1	-1	-1	-1	6	0	8	-1
1	-1	-1	4	-1	-1	-1	8	0	-1
5	-1	-1	2	-1	5	-1	-1	-1	-1

Routing or Connection Table:

	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10
N1		2	2	9	9	10	9	9	9	10
N2	1		3	3	3	1	3	1	1	1
N3	2	2		4	4	4	4	2	2	4
N4	9	3	3		5	10	5	9	9	10
N5	4	4	4	4		7	7	7	4	4
N6	10	10	10	10	7		7	7	10	10
N7	5	5	5	5	5	6		8	5	6
N8	9	9	9	9	7	7	7		9	7
N9	1	1	1	4	4	1	4	8		1
N10	1	1	4	4	4	6	6	6	4	

Shortest Path from Rooter: [4] to [10] is: 4 - 10
The total cost is 2

Shortest Path from Rooter: [2] to [8] is: 2 - 1 - 9 - 8
The total cost is 11

Shortest Path from Rooter: [10] to [1] is: 10 - 1
The total cost is 5

Similarly, tests have been performed for 10, 12, 15 and 20 Routers as well and the Test files have also been include in the Project Test_Files directory

Conclusion

The Link State Protocol has been simulated using Dijkstra's algorithm. The advantages of this protocol is that it has smaller, single route routing tables and has low network overhead once the internetwork has converged.

The disadvantages contained in this protocol is that it requires additional planning and configuration. For very large internetworks, the database of link state advertisements and the calculation of routing table entries can be memory and processor intensive.

Acknowledgement

We wish to express our sincere gratitude to Prof. Manikandan K., School of Computer Science and Engineering, VIT University, for encouraging us to take up this project and implement it. We sincerely thank him for his immense support and feedback provided through out the duration of this project.

REFERENCES

Cooke, K. L., and halsey, E. 1966. The shortest route through a network with time dependent internodal transit times. *J. Math. Anal. Appl*., 14, 493-498.

Dijkstra, E.W. A. 1959. note on two problems in connection with graphs. *Num. Anal*., 1 (Oct), 269-271.

Dreyfus, S.E. 1969. An appraisal of some shortest path algorithms. *Oper. Res*., 17, 395-412.

Even, S., and gazit, H. 1985, Updating distances in dynamic graphs. *Meth. Oper. Res*., 49 (May), 371-387.

Halpern, J. 1977. The shortest route with time dependent length of edges and limited delay possibilities in nodes. *Z. Oper. Res*., 21 (1977), 117-124.

Klafszky, E. 1972. Determination of shortest path in a network with time-dependent edge-lengths. *Math. Oper. Stat*., 3 255-257.

Ling, S. T., Furuno, K., and tezuka, Y.1972. Optimal path in networks with time-varying traverse time and expenses branches. *Tech. Rep. of Osaka University*, Japan.

Manikandan, K., Durai, M.S. and MA, S.K.D. 2011. Secure On-Demand Routing Protocol for MANET using Genetic Algorithm. *International Journal of Computer Applications*, pp.0975-8887.

Orda, A. and Rom, R. 1998. Distributed shortest-path protocols for time-dependent networks. In Proceedings of *lCCC,* 88 (Tel Aviv, Israel, Nov. pp. 439-445.

Orda, A., and Rom, R. 1989. Minimum weight paths in time-dependent networks. EE Publication No. 710. Faculty of *Electrical Engineering*, Technion, Haifa, Israel, Mar.

****