CS3205 Networks Assignment 3 Report

Author Name: **N K Sagar Reddy**

Roll no: **CS18B029**

Department of Computer Science and Engineering

April 20, 2021

Contents

1 Aim/objective

- The purpose of this lab is to implement a simplified version of the Open Shortest Path First (OSPF) routing protocol.

2 Introduction

- The OSPF (Open Shortest Path First) protocol is an IP Routing protocol, used to distribute IP routing information throughout a single a network
- The OSPF protocol is a link-state routing protocol, which means that the routers exchange topology information with their nearest neighbors.
- The topology information is flooded throughout the network, so that every router within the network has a complete picture of the topology of the network.
- The complete knowledge of topology allows routers to calculate routes that satisfy particular criteria(Fastest path,shortest path etc)

3 Experimental details

- We send a HELLO message from each node to its neighbors, once every HELLOINTERVAL seconds,the neighbors , upon receving the message give a random edge cost for that particular edge encoded in a HELLOREPLY message
- We also send a Link State Advertisement (LSA) message to its neighbors, once every LSAINTERVAL seconds. The neighbors upon receiving this LSA node transfer this LSA msg to all of their respective neighbors until all connected nodes receive this msg.
- We use sequence numbers to make sure infinite loops are messages are not sent in the given network of routers and also update the edges/transfer the packet only when the sequence number of the LSA Packet is greater than that of the previous LSA packet received from the same source
- When a node receives an LSA message from a neighbor, it will store the LSA information and forward the LSA to all interfaces other than the interface that the packet arrived on, if and only if the newly received LSA's sequence number is strictly greater than the last known sequence number from the sender.

3.1 Experimental/Simulation setup

- The input format is as follows, The first entry on the first line specifies the number of routers (N) The node indices go from 0 to (N-1) ,The second entry on the first line specifies the number of links M
- Each subsequent row contains the tuple (i, j, MinC_{ij} , MaxC_{ij}). This implies a bidirectional link between nodes i and j . The use of minimum and maximum is for generating a random number in between them.

- Each OSPF router (running as a Linux process) will perform the following actions:
  - Obtain necessary parameters including its node identifier (id) , Read the input file and find out its neighboring node identifiers.
  - Establish a UDP socket on port number (20001 + id) for all OSPF communications.
- Each OSPF router then Sends a HELLO message to its neighbors, once every HELLO INTERVAL seconds. This value is specified in the command line as x; Default: 1 second.
- Packet Format: HELLO srcid
- When a router receives a HELLO message on an interface, it will reply with the HELLOREPLY message along with the cost. The cost reported for link ij by node j for a packet received from node i is a RANDOM number between MinC_{ij} and MaxC_{ij} . Node i, on receiving this message, will store this value as the cost for link ij
- The message format is: HELLOREPLY j i costforlinkij
- Send a Link State Advertisement (LSA) message to its neighbors, once every LSA INTERVAL seconds. This value is specified in the command line.
- packet Format : LSA srcid seqno noofentries neigh1 cost1 neigh2 cost2 ...
- Using the LSA packets recieved from all other nodes , determine the network topology and use Dijkstra's algorithm to compute the shortest path and write it to the outfile.

3.2 Entities Involved

- id , it denotes the node no of the current router
- infile , the file from which input is read
- outfile , the file to which routing table is written to , every z seconds
- x , HELLO INTERVAL Time
- y , LSA INTERVAL time
- z , SPF INTERVAL time

3.3 Additional Details

- We use threads within a process to launch the Hello,LSA,Dijkstra functins parallely as they need to be called at certain time intervals independent of each other.
- We use locks in 2 functions/threads(one of which is Dijkstra) to ensure the graph/ any global variable used in the function does not get updated during the computation.

4 Results and Observations

- This section contains the output file of 2 nodes ,each one taken in a different session having a unique input file for that session(different topologies).
- Table-1 contains the output file of Node :0 when the input file is given as Input - 1
- Table-2 contains the output file of Node : 1 when the input file is given as Input - 2

Input-1:
8 22
0 1 4 10
1 2 3 9
2 0 6 10
3 1 4 10
3 2 3 9
0 3 6 10
0 4 2 5
4 1 7 20
2 4 3 7
4 3 9 17
0 5 10 15
5 1 13 20
2 5 20 27
5 3 25 26
0 6 12 16
6 1 13 17
2 6 4 6
6 3 1 5
0 7 9 15
7 1 15 20
2 7 19 24
7 3 30 35

Table - 1:
==========
Routing Table for Node number. 0 at Time 20

| Destination | Path | Cost |

	1	01	4
	2	02	9
	3	03	9
	4	04	3
	5	05	13
	6	036	12
	7	07	13

==========

Routing Table for Node number. 0 at Time 40

Destination	Path	Cost
1	01	7
2	02	10
3	03	10
4	04	5
5	05	15
6	06	15
7	07	14

==========

Input-2:

```
8  20
1  0   5  20
1  2  12  30
2  0  13  17
3  1  11  16
3  2  7  19
0  3  2  10
4  5  16  25
4  6  27  31
4  7  21  24
5  6  31  35
5  7  10  15
6  7  23  27
0  4  1  30
4  1  9  19
1  5  12  14
7  2  13  20
3  6  11  17
4  2  4  7
3  7  10  13
6  2  27  20
```

Table -2 :

==========

Routing Table for Node number. 1 at Time 20

Destination	Path	Cost
0	10	14
2	12	13
3	13	11
4	14	14
5	15	12
6	136	25
7	137	21

Routing Table for Node number. 1 at Time 40

Destination	Path	Cost
0	10	12
2	142	23
3	13	15
4	14	17
5	15	12
6	136	27
7	157	25

5 Learnings

- As we have a common global memory for each process which is accessed by all the threads , we need to use locks ,because of Data race conditions and also,the global variable should be constant to each thread unless the thread updates it.
- This kind of SPF computation at a regular interval helps the network to adapt to the dynamic state of the network as 2 routers always get the optimal path to use to exchange information.)

6 Additional Thoughts

As more number of routers are added,the frequency of exchange of LSA packets/topology updates increases, it even takes more time to calculate two end to end routes in constant intervals of time.

-

7 Conclusion

- We can use OSPF routing protocol to send messages between two routers which are not directly connected,by calculating the shortest paths regularly, with the network topology at that time and

send the message along the shortest path.

8 References

1)https://pythontic.com/modules/socket/udp-client-server-example
2)https://www.tutorialspoint.com/python3/python_multithreading.htm/
3)https://docs.python.org/3/library/_thread.html#:~:text=This%20module%20provides%20low%2Dlevel,or%20binary%20semaphores)%20are%20provided.