Networks Assignment 3

Nischith Shadagopan M N
CS18B102
Computer Science and Engineering
Indian Institute of Technology Madras

I. Aim

The aim of this assignment is to understand how routers route packets to their destination using the path with least delay.

II. Introduction

Choosing the path to pass information can be viewed as a policy decision. We can rout in such a way that we choose the shortest paths or we can route in a load balanced way and so on. In this assignment we will see how to route along the shortest paths using Dijkstra's shortest path algorithm.

III. Dijkstras shortest path algorithm

The shortest path is calculated as follows

notation

- c(i, j) : link cost from node i to j
- D(v) : current value of cost for path from source to node v
- p(v) : predecessor node along path from source to v
- S : set of nodes whose least cost path have been computed

pseudo code

```
S = {A}
for all nodes v:
    if v adjacent to A
        D(v) = c(A, v)
    else
        D(v) = ∞
while S doesnt contain all nodes:
    find w not in S such that D(w) is minimum
    add w to S
    for all v adjacent to w and not in S:
        if D(w) + c(w, v) < D(v):
            D(v) = D(w) + c(w, v)
            p(v) = w
```


This work was done as part of Introduction to Computer Networks course

IV. Implementational details

- Hello message is sent to its neighbours once every HELLO INTERVAL seconds.
- When a router receives a HELLO message it replies with the cost for the corresponding edge which is a random number between the limits specified in the input file.
- Link State Advertisement message is sent to neighbours once every LSA INTERVAL seconds.
- When a node receives LSA message it stores the info and forwards the message to its neighbours other than the one who sent it provided the sequence number is greater than the last received sequence number.
- Shortest paths are computed once every SPF INTERVAL seconds.
- Sending HELLO, sending LSA, computing shortest path and receiving messages are done on separate threads.

V. Results and Observations

For both the following cases default values of HELLO interval, LSA interval and SPF interval are used.
For the first test input lets keep min = max for all the edges so that the graph is fixed.

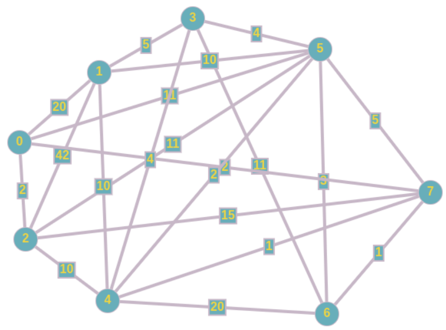
```
8 20
0 1 20 20
0 2 2 2
0 5 11 11
0 7 2 2
1 2 42 42
1 3 5 5
1 4 10 10
1 5 10 10
2 4 10 10
2 5 11 11
2 7 15 15
3 4 4 4
3 5 4 4
3 6 11 11
4 5 2 2
4 6 20 20
4 7 1 1
5 6 3 3
```

```
5 7 5 5
6 7 1 1
```

input.txt

This graph can be visualized as follows:

Fig. 1. Visualization of input graph



The routing tables computed for this input is as follows

```
Routing Table for Node No. 0 at Time 20
Destination                 Path            Cost
1                      0-7-4-3-1              12
2                            0-2               2
3                        0-7-4-3               7
4                          0-7-4               3
5                        0-7-4-5               5
6                          0-7-6               3
7                            0-7               2
```

Routing table for node 0

```
Routing Table for Node No. 1 at Time 20
Destination                 Path            Cost
0                      1-3-4-7-0              12
2                    1-3-4-7-0-2              14
3                            1-3               5
4                          1-3-4               9
5                          1-3-5               9
6                      1-3-4-7-6              11
7                        1-3-4-7              10
```

Routing table for node 1

```
Routing Table for Node No. 2 at Time 20
Destination                 Path            Cost
0                            2-0               2
1                    2-0-7-4-3-1              14
3                      2-0-7-4-3               9
4                        2-0-7-4               5
5                      2-0-7-4-5               7
6                        2-0-7-6               5
7                          2-0-7               4
```

Routing table for node 2

```
Routing Table for Node No. 3 at Time 20
Destination                 Path            Cost
0                        3-4-7-0               7
1                            3-1               5
2                      3-4-7-0-2               9
4                            3-4               4
5                            3-5               4
6                        3-4-7-6               6
7                          3-4-7               5
```

Routing table for node 3

```
Routing Table for Node No. 4 at Time 20
Destination                 Path            Cost
0                          4-7-0               3
1                          4-3-1               9
2                        4-7-0-2               5
3                            4-3               4
5                            4-5               2
6                          4-7-6               2
7                            4-7               1
```

Routing table for node 4

```
Routing Table for Node No. 5 at Time 20
Destination                 Path            Cost
0                        5-4-7-0               5
1                          5-3-1               9
2                      5-4-7-0-2               7
3                            5-3               4
4                            5-4               2
6                            5-6               3
7                          5-4-7               3
```

Routing table for node 5

```
Routing Table for Node No. 6 at Time 20
Destination                 Path            Cost
0                          6-7-0               3
1                      6-7-4-3-1              11
2                        6-7-0-2               5
3                        6-7-4-3               6
4                          6-7-4               2
5                            6-5               3
7                            6-7               1
```

Routing table for node 6

```
Routing Table for Node No. 7 at Time 20
Destination                 Path            Cost
0                            7-0               2
1                        7-4-3-1              10
2                          7-0-2               4
3                          7-4-3               5
4                            7-4               1
5                          7-4-5               3
6                            7-6               1
```

Routing table for node 7

In this test case the graph is fixed and the shortest paths doesn't change with time. One can verify that the program is correctly computing the shortest paths in this case. This means that all the nodes are correctly obtaining graph information through LSA packets and are correctly computing shortest paths.

For the second test case lets choose a dynamically changing graph

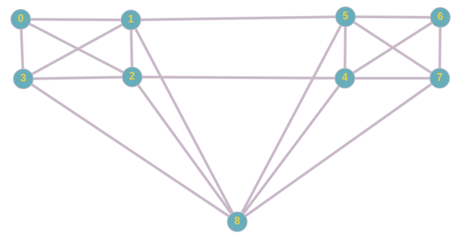
```
9 20
0 1 1 3
0 2 4 5
0 3 2 6
1 2 5 7
1 3 3 9
1 5 4 7
1 8 2 7
2 3 4 5
2 4 7 9
2 8 2 7
3 8 2 7
4 5 5 8
4 6 2 6
```

```
4 7 2 8
4 8 5 7
5 6 1 5
5 7 2 7
5 8 4 9
6 7 4 7
7 8 2 8
```

input.txt

Lets visualize this graph

Fig. 2. Visualization of input graph



Lets see the routing tables for this input graph

```
Routing Table for Node No. 0 at Time 20
Destination              Path           Cost
1                         0-1              2
2                         0-2              4
3                         0-3              5
4                       0-2-4             11
5                       0-1-5              7
6                     0-1-5-6              9
7                     0-1-5-7             10
8                       0-1-8              5
Routing Table for Node No. 0 at Time 40
Destination              Path           Cost
1                         0-1              2
2                         0-2              4
3                         0-3              2
4                       0-2-4             11
5                       0-1-5              6
6                     0-1-5-6              9
7                     0-1-5-7              9
8                       0-1-8              6
```

Routing table for node 0

```
Routing Table for Node No. 1 at Time 20
Destination              Path           Cost
0                         1-0              2
2                         1-2              6
3                       1-0-3              5
4                       1-5-4             10
5                         1-5              4
6                       1-5-6              6
7                       1-5-7              7
8                         1-8              7
Routing Table for Node No. 1 at Time 40
Destination              Path           Cost
0                         1-0              2
2                         1-2              6
3                       1-0-3              5
4                       1-5-4              9
5                         1-5              4
6                       1-5-6              7
7                       1-5-7              7
8                         1-8              4
```

Routing table for node 1

```
Routing Table for Node No. 2 at Time 20
Destination              Path           Cost
0                         2-0              5
1                         2-1              6
3                         2-3              5
4                         2-4              8
5                       2-8-5             11
6                     2-8-5-6             13
7                       2-8-7             11
8                         2-8              4
Routing Table for Node No. 2 at Time 40
Destination              Path           Cost
0                         2-0              5
1                         2-1              6
3                         2-3              4
4                         2-4              7
5                       2-1-5             10
6                       2-4-6              9
7                     2-1-5-7             13
8                         2-8              7
```

Routing table for node 2

```
Routing Table for Node No. 3 at Time 20
Destination              Path           Cost
0                         3-0              3
1                       3-0-1              5
2                         3-2              5
4                       3-8-4             11
5                     3-0-1-5             10
6                   3-0-1-5-6             12
7                       3-8-7             11
8                         3-8              4
Routing Table for Node No. 3 at Time 40
Destination              Path           Cost
0                         3-0              3
1                       3-0-1              5
2                         3-2              4
4                       3-8-4              9
5                       3-8-5              9
6                     3-8-4-6             11
7                       3-8-7              9
8                         3-8              2
```

Routing table for node 3

```
Routing Table for Node No. 4 at Time 20
Destination              Path           Cost
0                   4-6-5-1-0             11
1                     4-6-5-1              9
2                         4-2              7
3                       4-2-3             12
5                       4-6-5              4
6                         4-6              2
7                         4-7              7
8                         4-8              7
Routing Table for Node No. 4 at Time 40
Destination              Path           Cost
0                   4-6-5-1-0             11
1                     4-6-5-1              9
2                         4-2              7
3                       4-2-3             11
5                       4-6-5              5
6                         4-6              2
7                         4-7              7
8                         4-8              7
```

Routing table for node 4

```
Routing Table for Node No. 5 at Time 20
Destination              Path           Cost
0                       5-1-0              7
1                         5-1              5
2                       5-1-2             11
3                       5-8-3             10
4                         5-4              8
6                         5-6              3
```

```
7                       5-7              3
8                       5-8              5
Routing Table for Node No. 5 at Time 40
Destination            Path          Cost
0                     5-1-0              6
1                       5-1              4
2                     5-1-2             10
3                   5-1-0-3              9
4                       5-4              5
6                       5-6              3
7                       5-7              3
8                       5-8              7
```

Routing table for node 5

```
Routing Table for Node No. 6 at Time 20
Destination            Path          Cost
0                   6-5-1-0              9
1                     6-5-1              7
2                     6-4-2              9
3                 6-5-1-0-3             12
4                       6-4              2
5                       6-5              2
7                     6-5-7              5
8                     6-4-8              9
Routing Table for Node No. 6 at Time 40
Destination            Path          Cost
0                   6-5-1-0              9
1                     6-5-1              7
2                     6-4-2              9
3                 6-5-1-0-3             12
4                       6-4              2
5                       6-5              3
7                       6-7              6
8                     6-4-8              9
```

Routing table for node 6

```
Routing Table for Node No. 7 at Time 20
Destination            Path          Cost
0                   7-5-1-0             10
1                     7-5-1              8
2                     7-4-2             14
3                     7-8-3             12
4                       7-4              7
5                       7-5              3
6                     7-5-6              6
8                       7-8              7
Routing Table for Node No. 7 at Time 40
Destination            Path          Cost
0                   7-5-1-0              9
1                     7-5-1              7
2                   7-5-1-2             13
3                     7-8-3             12
4                       7-4              7
5                       7-5              3
6                     7-5-6              6
8                       7-8              7
```

Routing table for node 7

```
Routing Table for Node No. 8 at Time 20
Destination            Path          Cost
0                     8-1-0              5
1                       8-1              3
2                       8-2              7
3                       8-3              5
4                       8-4              7
5                       8-5              7
6                     8-5-6              9
7                       8-7              7
Routing Table for Node No. 8 at Time 40
Destination            Path          Cost
0                     8-1-0              6
1                       8-1              4
2                       8-2              7
3                       8-3              5
4                       8-4              7
5                       8-5              7
6                     8-4-6              9
7                       8-7              7
```

Routing table for node 8

Firstly we can notice that the shortest paths are changing with time. This is because the graph changes dynamically due to the random number return in HELLOREPLY messages. This simulates dynamically changing costs of sending packets on a link depending on congestion.
Also each node in the graph might have different views of the graph while computing shortest paths. For example the path cost from 0-2 is 4 but path cost from 2-0 is 5 even though the graph is undirected. This is due to two reasons. One is that when we receive a HELLOREPLY message we update the cost for the edge in this node. But this change is not known to other nodes until they receive a LSA message. If we compute shortest path before we receive the LSA message then the nodes will have different views of the graph. Another reason is we are using UDP for all communication and UDP is not reliable. If a LSA packet is lost then that node will not receive information about the graph and this can lead to different views of the graph.
Also if $ospf > hi > lsai$ then there are a lot more lsa packets sent before the graph changes and hence there is a better chance of all the nodes having a common view of the graph.

VI. Learning Outcomes

- Clear understanding of how routers build routing tables with the aim of choosing shortest paths
- Saw how it computes routing table as the network changes dynamically
- Saw how routers gather information about the network through HELLO and LSA messages.
- Learnt to write script files
- Learnt to document my work thoroughly

VII. Additional thoughts

Another way of implementing this is using only LSA messages to compute shortest paths. Information obtained through HELLOREPLY messages is not used for shortest path computation, it is only used for sending LSA messages. This way there is higher chance of nodes having the same view of the graph. But they can still have different views of the graph if the LSA packets are dropped.

VIII. Conclusion

We can see how nodes obtain information about the graph by flooding LSA messages and through

HELLO messages. We have also simulated changing cost of sending packets through links due to varying congestion at the links. We saw how routers compute routing tables in this dynamically changing graph.

REFERENCES

[1] Vipin, Assignment 3 problem statement pdf and slides.