Atharva Yadav
S23-127

Network Lab

NETWORK LAB ASSIGNMENT NO.5B

Aim:
To simulate and study the link state routing algorithm using simulation using NS2.

Link state routing algorithm is a type of routing algorithm used in computer networks to determine the shortest path from a source node to all other nodes in the network.

Each node in the network maintains a map of the network topology, which includes information about all links and their states (i.e., whether they are up or down).

Each node floods its link state information to all other nodes in the network.

Upon receiving this information, each node constructs a complete map of the network topology.

Using this topology map, each node computes the shortest path to all other nodes using algorithms like Dijkstra's algorithm.

Finally, each node uses this shortest path information to forward packets towards their destination.

CODE:

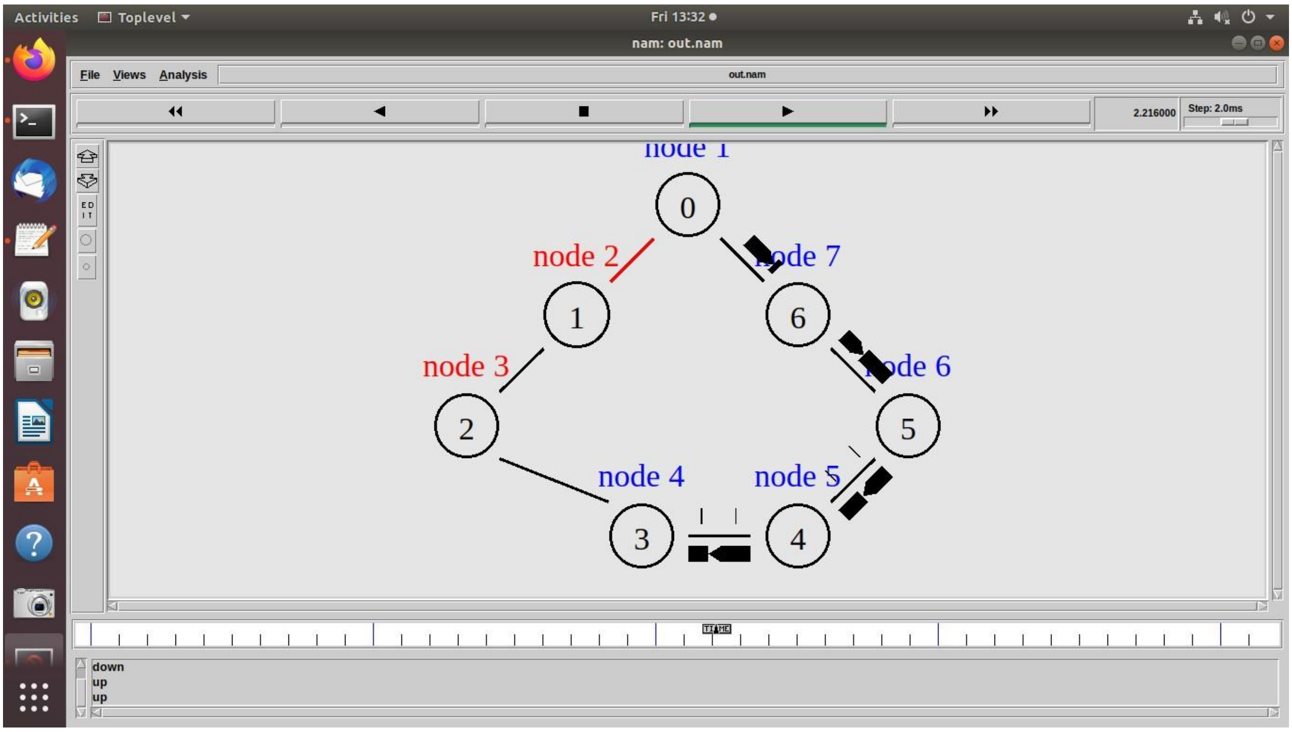
```
set tf [open out.tr w]
$ns trace-all $tf set nf
[open out.nam w] $ns
namtrace-all $nf
$node1 label "node 1"
$node2 label "node 2"
$node3 label "node 3"
$node4 label "node 4"
$node5 label "node 5"
$node6 label "node 6"
$node7 label "node 7"
$node1 label-color blue
$node2 label-color red

$node3 label-color red
$node4 label-color blue
$node5 label-color blue
$node6 label-color blue
$node7 label-color blue
$ns duplex-link $node1 $node2 1.5Mb 10ms DropTail
$ns duplex-link $node2 $node3 1.5Mb 10ms DropTail
```

```
$ns duplex-link $node3 $node4 1.5Mb 10ms DropTail
$ns duplex-link $node4 $node5 1.5Mb 10ms DropTail
$ns duplex-link $node5 $node6 1.5Mb 10ms DropTail
$ns duplex-link $node6 $node7 1.5Mb 10ms DropTail
$ns duplex-link $node7 $node1 1.5Mb 10ms DropTail
$ns duplex-link-op $node1 $node2 orient left-down
$ns duplex-link-op $node2 $node3 orient left-down
$ns duplex-link-op $node3 $node4 orient right-down
$ns duplex-link-op $node4 $node5 orient right
$ns duplex-link-op $node5 $node6 orient right-up
$ns duplex-link-op $node6 $node7 orient left-up
$ns duplex-link-op $node7 $node1 orient left-up
```

```
set tcp2 [new Agent/TCP] $ns attach-
agent $node1 $tcp2 set sink2 [new
Agent/TCPSink] $ns attach-agent
$node4 $sink2 $ns connect $tcp2
$sink2 set traffic_ftp2 [new
Application/FTP] $traffic_ftp2 attach-
agent $tcp2 proc finish {} { global ns
nf $ns flush-trace close $nf exec nam
out.nam & exit 0
}
$ns at 0.5 "$traffic_ftp2 start"
$ns rtmodel-at 1.0 down $node1 $node2
$ns rtmodel-at 2.0 up $node2 $node3
$ns at 3.0 "$traffic_ftp2 start"
$ns at 4.0 "$traffic_ftp2 stop"
$ns at 5.0 "finish"
$ns run
```

OUTPUT:



Conclusions: Link-state protocols like OSPF are suitable for large, dynamic networks with high traffic. Their fast convergence and scalability make them ideal for complex network topologies. Consider network size, traffic patterns, and reliability requirements when selecting a routing protocol for simulation.

LO: Based on LO4 : To implement client-server socket programs.