NETWORK PROTOCOLS & SECURITY
23EC2210 R/A/E

Topic:

ROUTING ALGORITHMS: DISTANCE VECTOR, LINKSTATE, HIERARCHICAL

Session - 21

AIM OF THE SESSION

To familiarize students with the Dynamic routing algorithms

INSTRUCTIONAL OBJECTIVES

This Session is designed to:

1. Describe the routing process of Distance Vector algorithm
2. Describe the routing process of Link state routing algorithm
3. Describe the routing process of Hierarchical routing algorithm

LEARNING OUTCOMES

At the end of this session, you should be able to:

1. Compute or update a router’s routing table using distance vector routing algorithm.
2. Explain the drawback of Distance vector routing.
3. Illustrate the working of Link state routing algorithm.
4. Illustrate the working of Hierarchical routing algorithm.

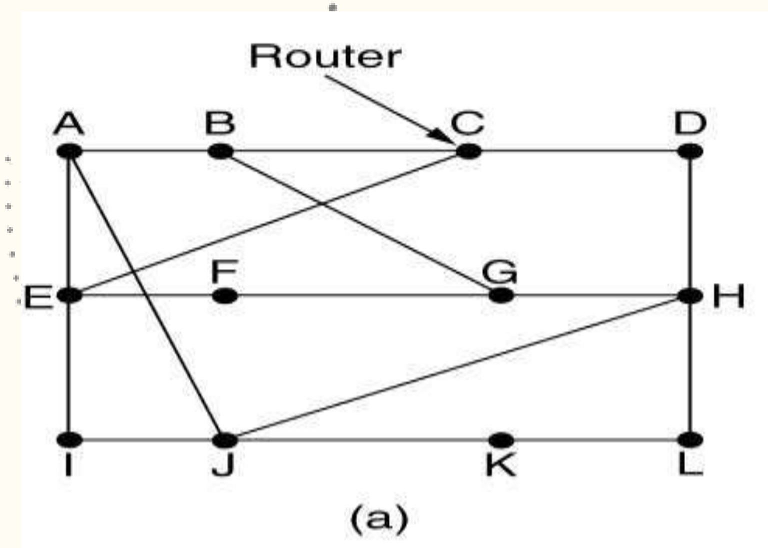
DISTANCE VECTOR ROUTING

DISTANCE VECTOR ROUTING

- Dynamic routing algorithm.
- More efficient because they find shortest paths for the current topology.
- Popular dynamic algorithms: Distance vector routing and Link state routing

- A **distance vector routing** algorithm operates by having each router maintain a table (i.e., a vector) giving the best-known distance to each destination and which link to use to get there.
- These tables are updated by exchanging information with the neighbors. Eventually, every router knows the best link to reach each destination.
- In distance vector routing, each router maintains a routing table indexed by, and containing one entry for each router in the network.
- Each table entry has two parts:
    - ➢ The preferred outgoing line to use for that destination
    - ➢ An estimate of the distance to that destination.
- The distance might be measured as the number of hops or using another metric

Distance Vector Routing…



To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6	New routing table for J	

Vectors received from J's four neighbors

(b)

Each node knows the distance (=cost) to its directly connected neighbours
– A node sends a list to its neighbors with the current distances to all nodes
– If all nodes update their distances, the routing tables eventually converge

(a) A subnet.

(b) Input from A, I, H, K, and the new routing table for J.

Count-to-Infinity problem:

- Drawback of distance vector routing is “**Count-to-Infinity**” problem.

- Distance vector routing is useful as a simple technique by which routers can collectively compute shortest paths.

- In particular, it reacts rapidly to good news, but leisurely to bad news.

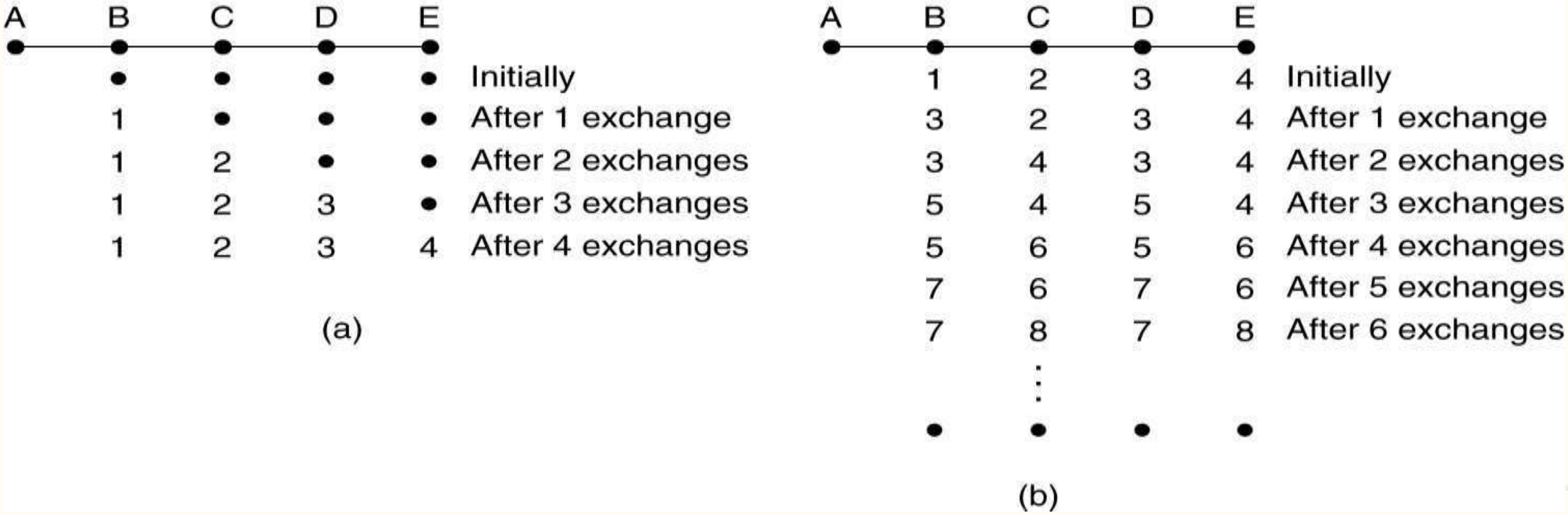


Fig: The count-to-infinity problem.

LINK STATE ROUTING

LINK STATE ROUTING

- The idea behind link state routing is simple.

- Each router must do the following five steps:
    1. Discover its neighbors, learn their network address.
    2. Measure the delay or cost to each of its neighbors.
    3. Construct a packet telling all it has just learned.
    4. Send this packet to all other routers.
    5. Compute the shortest path to every other router.

- In effect, the complete topology is distributed to every router. Then Dijkstra's algorithm can be run at each router to find the shortest path to every other router.

Link State Routing…

1. Learning about the Neighbors:

- When a router is booted, its first task is to learn who its neighbors are.
- It accomplishes this goal by sending a special HELLO packet on each point-to-point line.
- The router on the other end is expected to send back a reply giving its name.
- These names must be globally unique.

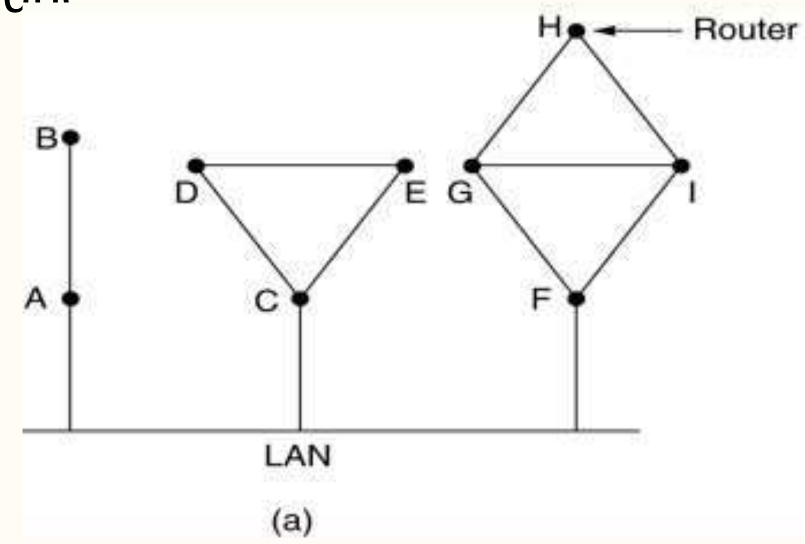


Fig. a Nine routers and a LAN.

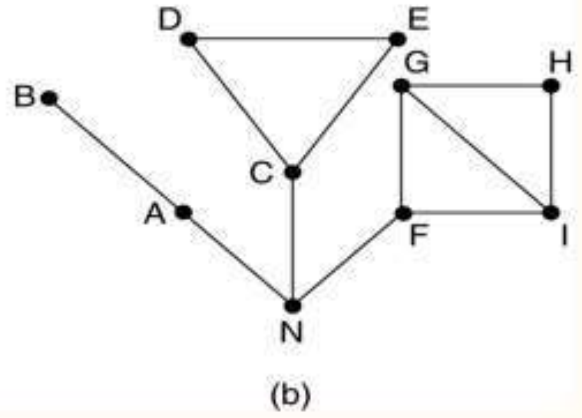


Fig. b A graph model of (a).

Link State Routing…

2. Measuring line cost:

- The link state routing algorithm requires each link to have a distance or delay for finding shortest paths.

- The most direct way to determine this delay is to send over the line a special ECHO packet that the other side is required to send back immediately.

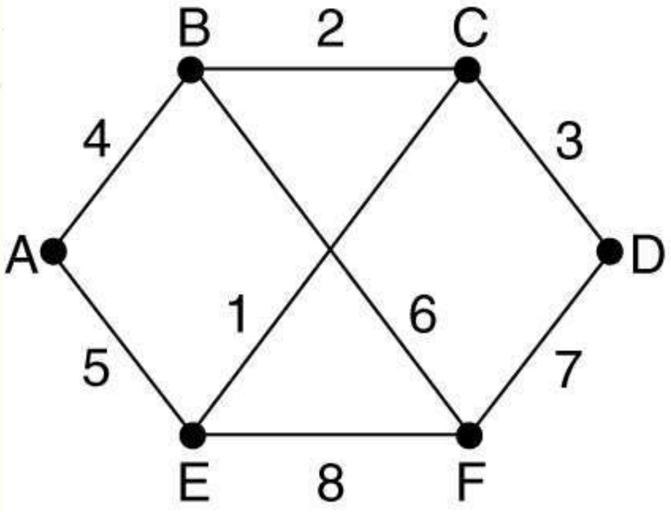
- By measuring the round-trip time and dividing it by two, the sending router can get a reasonable estimate of the delay.

Link State Routing…

3. Building Link State Packets:

- Once the information needed for the exchange has been collected, the next step is for each router to build a packet containing all the data.
- The packet contains:
  - ➢ The identity of the sender
  - ➢ Sequence number
  - ➢ Age
  - ➢ List of neighbors and cost to each neighbor
- Building the link state packets is easy.
- The hard part is determining when to build them. One possibility is to build them periodically at regular intervals.
- Another possibility is to build them when some significant event occurs, such as a line or neighbor going down or coming back up again or changing its properties appreciably.

Building Link State Packets



(a)

Link State Packets

A		B		C		D		E		F	
Seq.		Seq.		Seq.		Seq.		Seq.		Seq.	
Age		Age		Age		Age		Age		Age	
B	4	A	4	B	2	C	3	A	5	B	6
E	5	C	2	D	3	F	7	C	1	D	7
		F	6	E	1			F	8	E	8

(b)

(a) A subnet. (b) The link state packets for this subnet.

Link State Routing…

4. Distributing the Link State Packets:

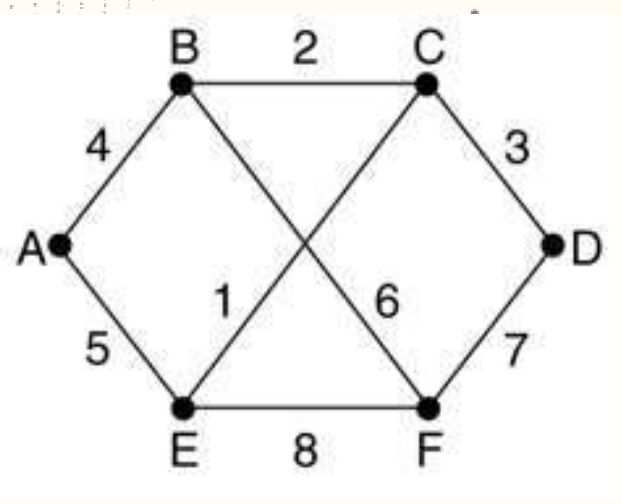
- The trickiest part of the algorithm is distributing the link state packets. All of the routers must get all of the link state packets quickly and reliably.

- When a link state packet comes into a router for flooding, it is not queued for transmission immediately. Instead, it is put in a holding area to wait a short while in case more links are coming up or going down.

- If another link state packet from the same source comes in before the first packet is transmitted, their sequence numbers are compared. If they are equal, the duplicate is discarded. If they are different, the older one is thrown out.

- To guard against errors on the links, all link state packets are acknowledged.

Distributing the Link State Packets



Source	Seq.	Age	Send flags A	Send flags C	Send flags F	ACK flags A	ACK flags C	ACK flags F	Data
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

Fig: The packet buffer for router B

- The data structure used by router *B* for the network shown above.
- Each row here corresponds to a recently arrived, but as yet not fully processed, link state packet.
- The table records where the packet originated, its sequence number and age, and the data.
- In addition, there are send and acknowledgement flags for each of *B*’s three links (to *A*, *C*, and *F*, respectively).

Link State Routing…

5. Computing the New Routes:

- Once a router has accumulated a full set of link state packets, it can construct the entire network graph because every link is represented.

- Now Dijkstra’s algorithm can be run locally to construct the shortest paths to all possible destinations.

- The results of this algorithm tell the router which link to use to reach each destination. This information is installed in the routing tables, and normal operation is resumed.

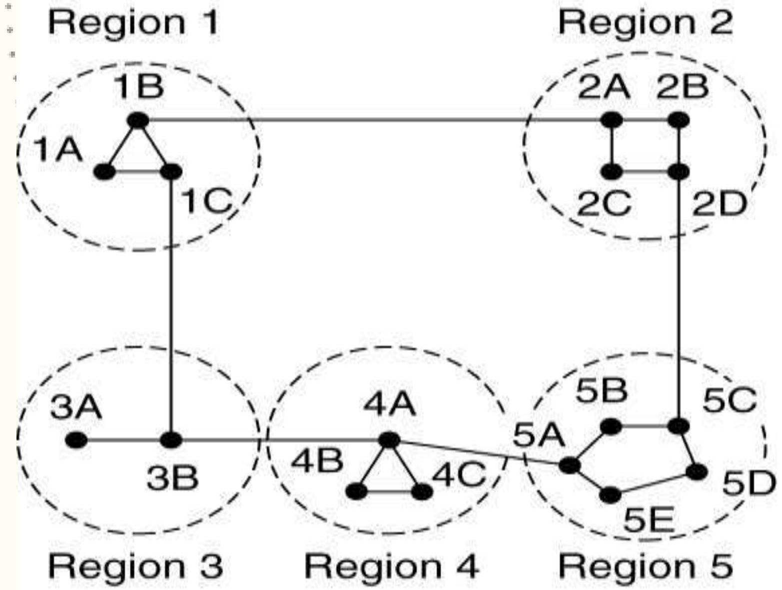
HIERARCHICAL ROUTING

HIERARCHICAL ROUTING

- As networks grow in size, the router routing tables grow proportionally.
- Due to which the memory consumption, CPU time is increased and requires more bandwidth to send status reports about them.
- At a certain point, the network may grow to the point where it is no longer feasible for every router to have an entry for every other router.
- Solution is **Hierarchical routing.**
- When hierarchical routing is used, the routers are divided into **regions**.
- Each router knows all the details about how to route packets to destinations within its own region but knows nothing about the internal structure of other regions.
- When different networks are interconnected, it is natural to regard each one as a separate region to free the routers in one network from having to know the topological structure of the other ones.

Hierarchical Routing...

Hierarchical routing.



(a)

Full table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

SUMMARY

- Distance Vector Routing
- Count-to-Infinity Problem
- Link State Routing
- Hierarchical Routing

TERMINAL QUESTIONS

1. Describe the steps of Link State routing algorithm.

2. Define Count-to-Infinity problem.

3. Explain the process of computing routing table using Distance vector routing algorithm.

REFERENCES FOR FURTHER LEARNING OF THE SESSION

Reference Books:

1. A.S. Tanenbaum, David J. Wetheral “Computer Networks” Pearson, 5th Edition.
2. Behrouz A. Forouzan , “Data Communication and Networking”, TMH, 5th Edition, 2012.

Web Links:

1. https://www.tutorialspoint.com/the-optimality-principle-in-computer-networks

THANK YOU

(DEEMED TO BE UNIVERSITY)

Team - Network Protocols & Security