Lecture-4

The Network Layer

Dr. Md. Imdadul Islam
Professor, Department of Computer Science and Engineering
Jahangirnagar University
https://www.juniv.edu/teachers/imdad

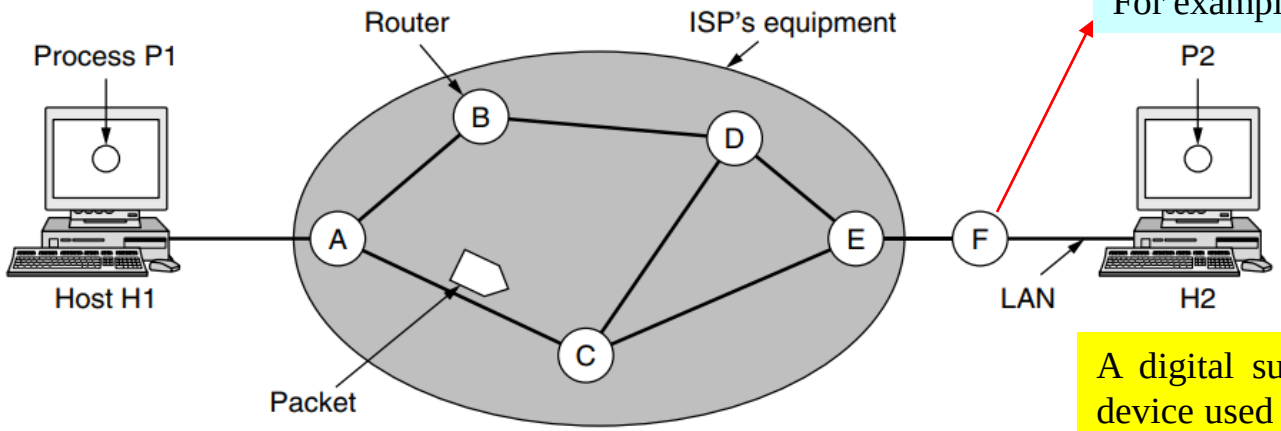
Network Layer Design Isues

- Store-and-Forward Packet Switching
- Services Provided to the Transport Layer
- Implementation of Connectionless Service
- Implementation of Connection-Oriented Service
- Comparison of Virtual-Circuit and Datagram Subnets
- Different routing algorithms

Store-and-Forward Packet Switching

The major components of the network are the ISP's equipment or carrier equipment (routers connected by transmission lines), shown inside the shaded oval, and the customers' equipment, shown outside the oval. Host H1 is directly connected to one of the ISP's routers, A, perhaps as a home computer that is plugged into a DSL modem.

For example Jagangirnagar University Router



The environment of the network layer protocols.

A digital subscriber line (DSL) modem is a device used to connect a computer or router to a telephone line which provides connection to Internet, which is often called DSL broadband.

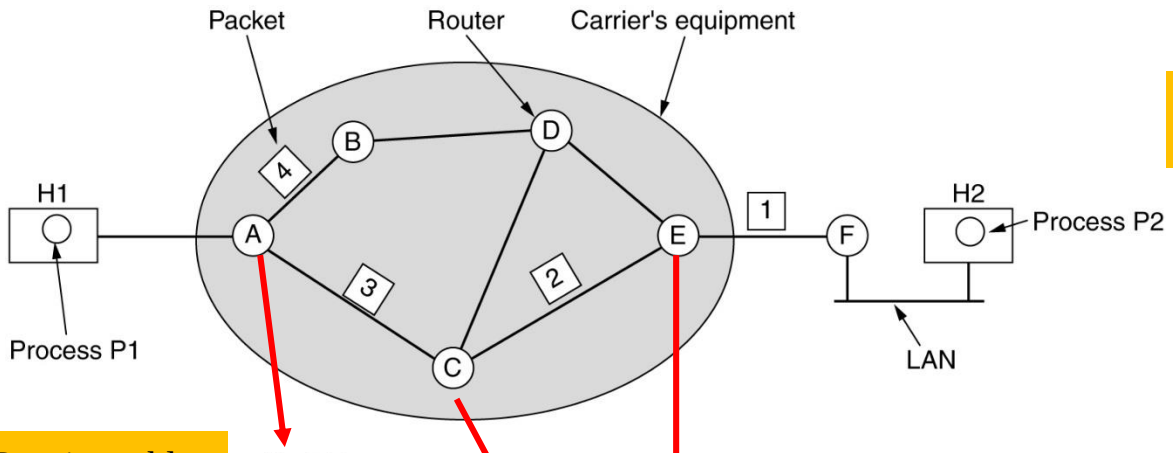
In contrast, H2 is on a LAN, which might be an office Ethernet, with a router, F, owned and operated by the customer. This router has a leased line to the ISP's equipment. We have shown F as being outside the oval because it does not belong to the ISP.

Implementation of Connectionless Service

✓If connectionless service is offered, packets are injected onto the subnet individually and routed independently of each other. No advanced setup is needed. In this context, the packet are frequently called **datagram's** (in analogy with telegram) and the network is called a datagram network.

✓If connection oriented service is used, a path from the source to destination router must be established before any data packets can be sent. This connection is called a VC (**virtual circuit**), in analogy with the physical circuits set up by the telephone system, and the network is called a virtual circuit network.

Suppose that the process P1 in fig. has a long message for P2. It hand the message to the transport layer with instructions to deliver it to process P2 on host H2. The transport layer code runs on H1, typically within the operating system. It prepends (add something) a transport header to the front of the message and hands the result to the network layer, probably just another procedure within the operating system.



Connectionless Service

Routing tables

A's table

Destination	Line (initially)	Destination	Line (later)
A	–	A	–
B	B	B	B
C	C	C	C
D	B	D	B
E	C	E	B
F	C	F	B

C's table

Destination	Via which node
A	A
B	A
C	–
D	D
E	E
F	E

E's table

Destination	Via which node
A	C
B	D
C	C
D	D
E	–
F	F

The message is four times longer than the maximum packet size, so the network layer has to break the it into four packets 1, 2, 3 and 4 and sends each of them in turn to router A using some point-to-point protocol.

Destination

Via which node

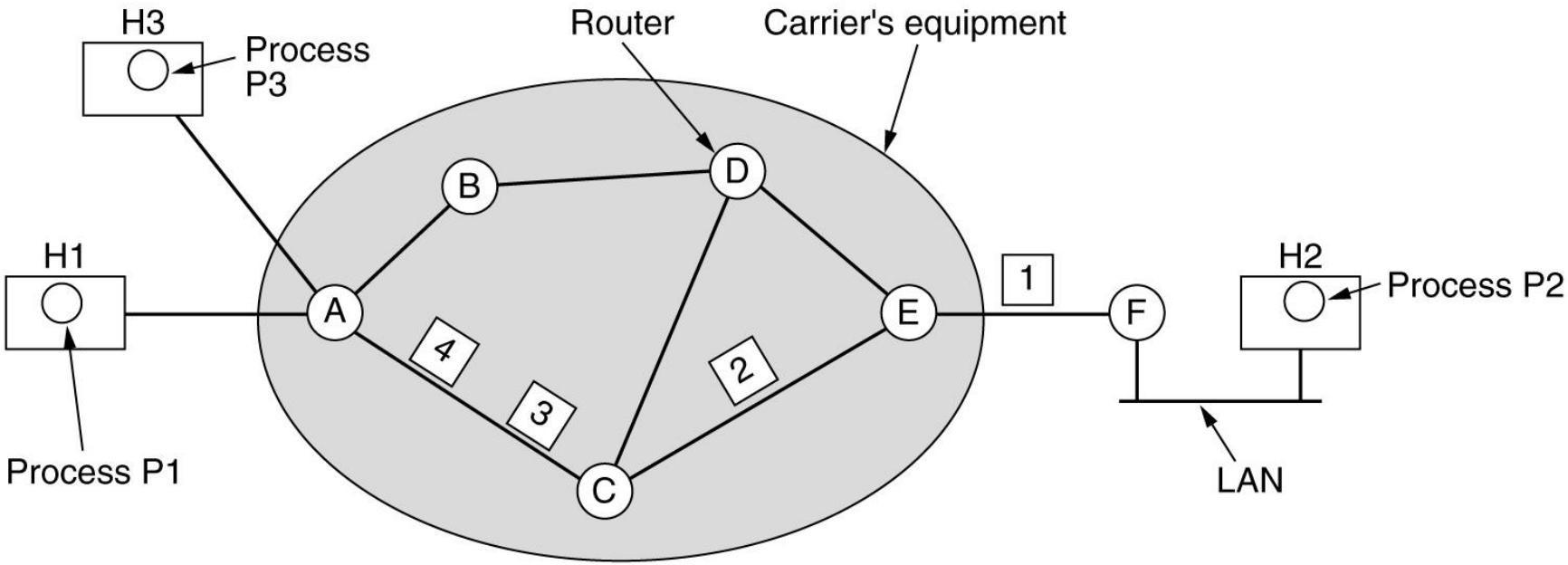
Routing of datagram in a network

Implementation of Connection-Oriented Service

- ✓ A fixed route is used for all traffic flowing over the connection, exactly the same way that the telephone system works. When the connection is released, the virtual circuit is also terminated.
- ✓ With connection-oriented service, each packet carries an identifier telling which virtual circuit it belongs to.

Implementation of connection-oriented service

Let us consider the connection between : (i) host H1 to H2 (ii) Host H3 to H2



A's table

In		Out	
H1	1	C	1
H3	1	C	2

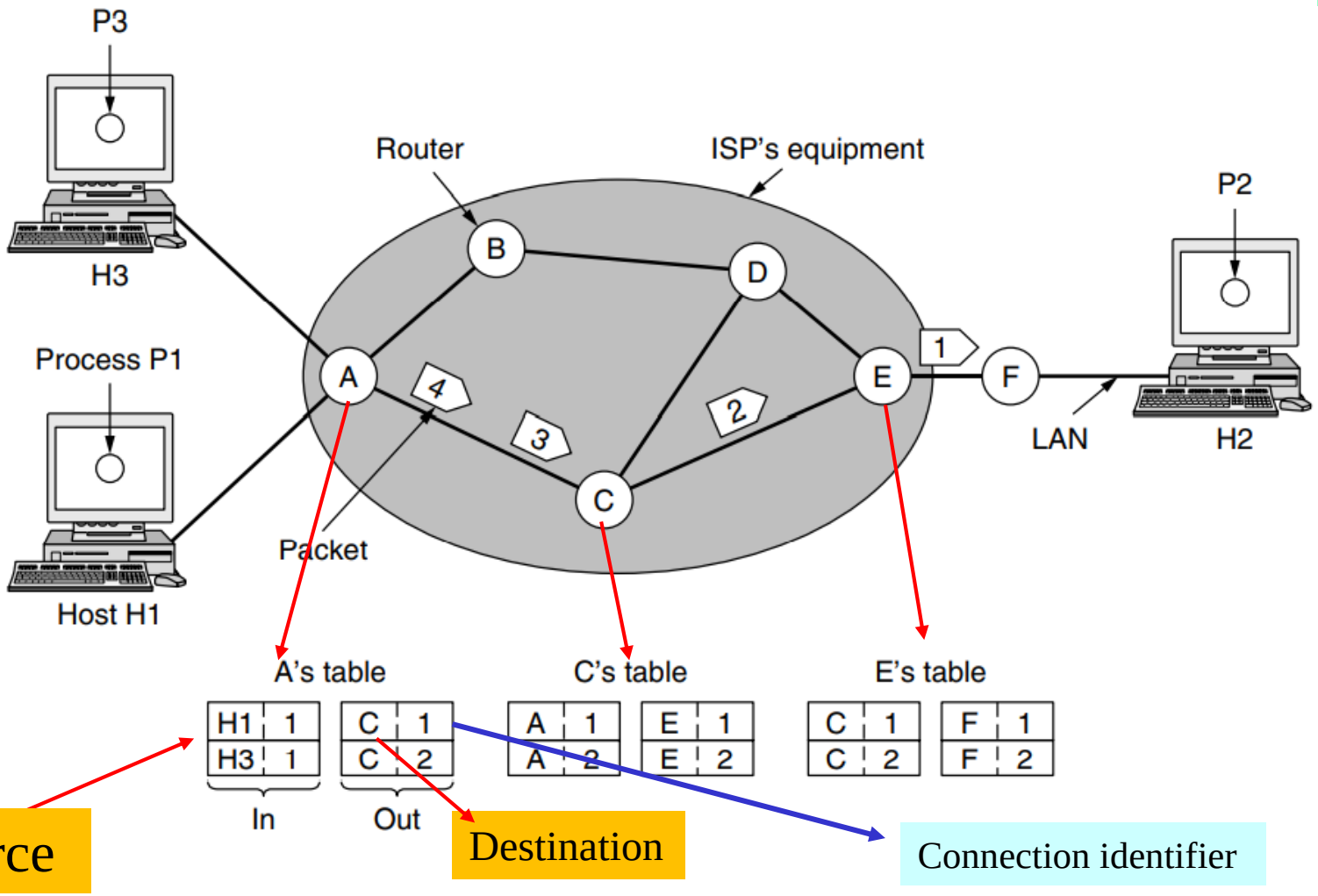
C's table

A	1	E	1
A	2	E	2

E's table

C	1	F	1
C	2	F	2

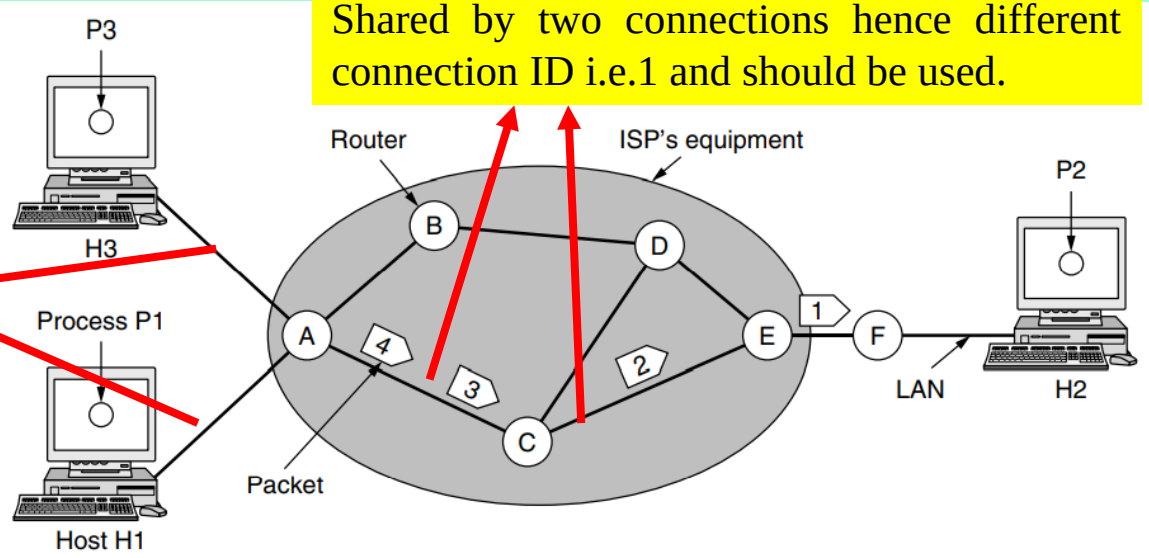
- ✓ Here, host H1 has established connection 1 with host H2. This connection is remembered as the first entry in each of the routing tables.
- ✓ The first line of A's table says that if a packet bearing **connection identifier 1** comes in from H1, it is to be sent to router C and given **connection identifier 1**. Similarly, the first entry at C routes the packet to E, also with **connection identifier 1**.



- ✓ Now let us consider what happens if H3 also wants to establish a connection to H2. It chooses **connection identifier 1** (because it is initiating the connection, and this is its only connection between H3 and A) and tells the network to establish the virtual circuit. This leads to the second row in the tables.
- ✓ Note that we have a conflict here because although A can easily distinguish connection 1 packets from H1 from connection 1 packets from H3, C cannot do this. For this reason, A assigns a different connection identifier to the outgoing traffic for the second connection.
- ✓ Avoiding conflicts of this kind is why routers need the ability to replace connection identifiers in outgoing packets.

Shared by two connections hence different connection ID i.e.1 and should be used.

Not shared by other connection hence same connection ID i.e.1 can be used.



A's table

H1	1	C	1
H3	1	C	2

In Out

C's table

A	1	E	1
A	2	E	2

E's table

C	1	F	1
C	2	F	2

Comparison of Virtual-Circuit and Datagram Subnets

Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

Routing Algorithms

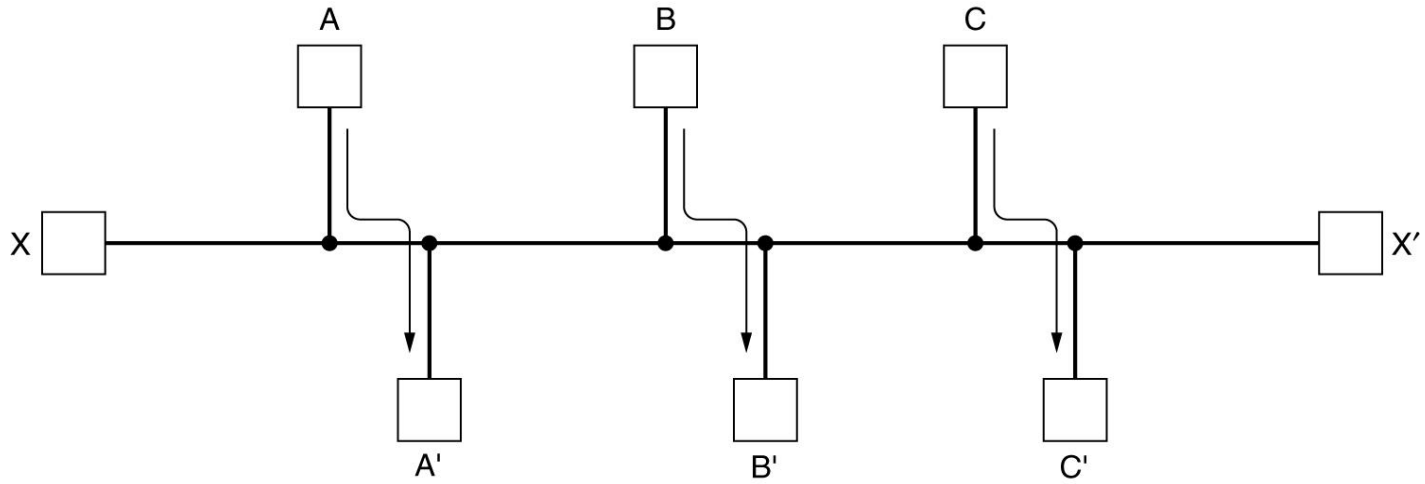
- The Optimality Principle
- Shortest Path Routing
- Flooding
- Distance Vector Routing
- Link State Routing
- Hierarchical Routing
- Broadcast Routing
- Multicast Routing
- Routing for Mobile Hosts
- Routing in Ad Hoc Networks

The routing algorithm is that part of the network layer software responsible for deciding which output line an incoming packet should be transmitted on. Certain properties are desirable in a routing algorithm: correctness, simplicity, robustness, stability, fairness and optimality.

Nonadaptive algorithms (static routing) do not base their routing decisions on any measurements or estimates of the current topology and traffic.

Adaptive algorithms, in contrast, change their routing decisions to reflect changes in the topology, and sometimes changes in the traffic as well.

Fairness and optimally may sound obvious-surely no reasonable person would oppose them-but as it turns out, they often contradictory goals. Suppose that there is enough traffic between A and A', between B and B', and between C and C' to saturate the horizontal links. To maximize the total flow, the X and X' traffic should be shut off altogether. Unfortunately, X and X' nay not see it that way. Evidently some compromise between global efficiency and fairness to individual connections is needed.

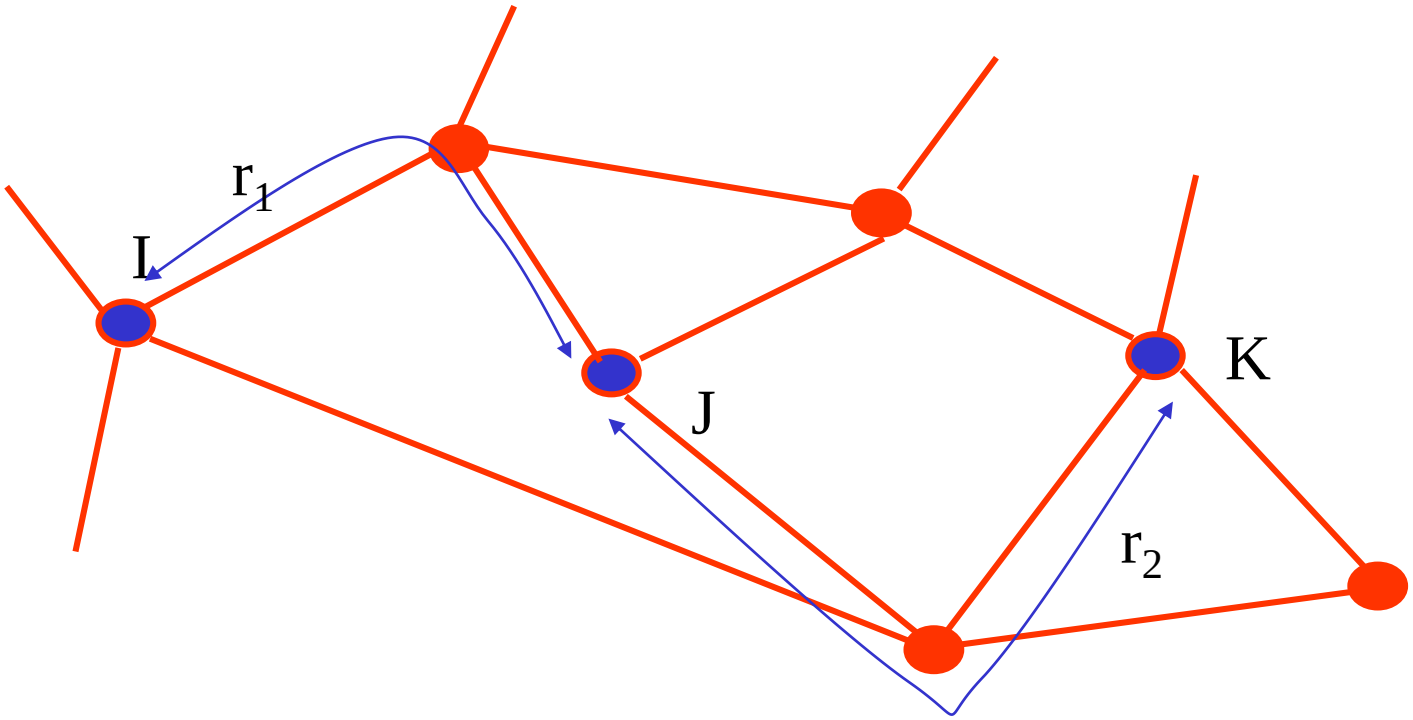


Conflict between fairness and optimality.

In context of **optimality**, we actually need to reduce delay of packet. The routing algorithm attempt to minimize the number of hops to reduce delay and increase the throughput.

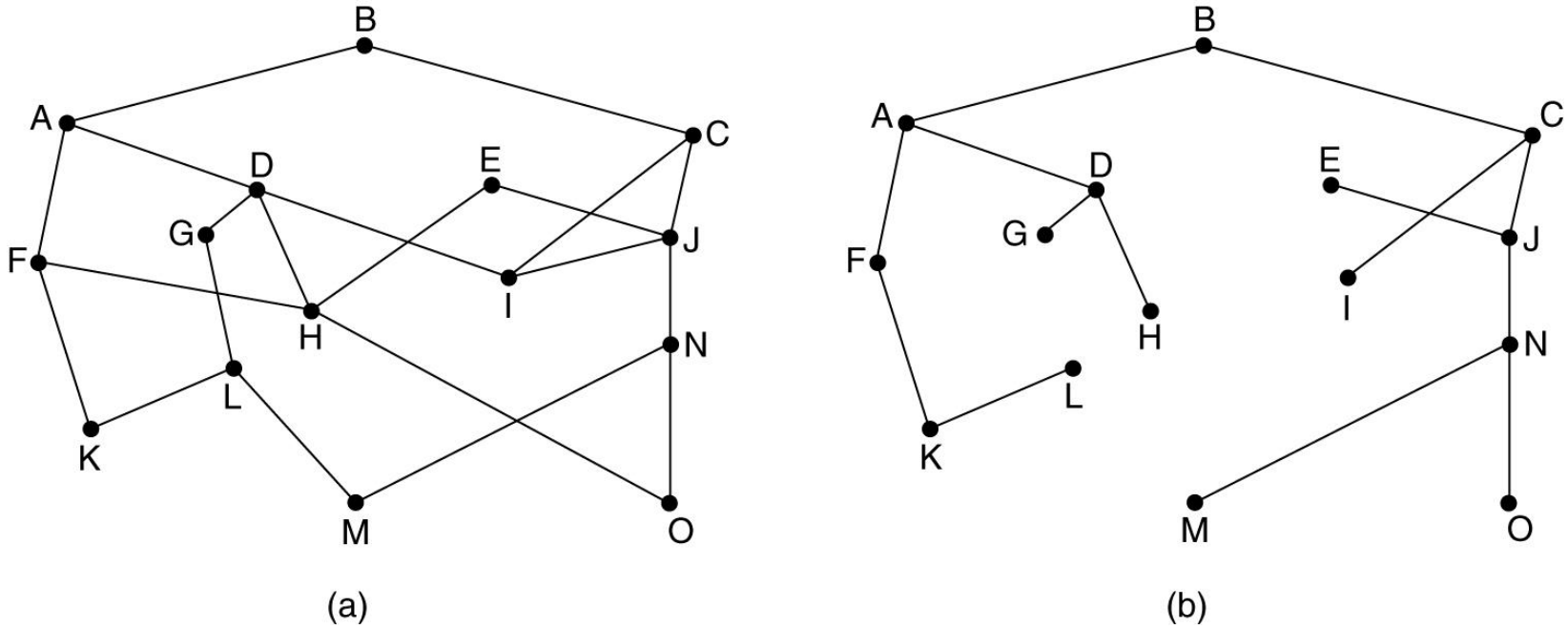
The Optimality Principle

- ✓ It states that if router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route.
- ✓ To see this, call the part of the route from I to J is r_1 and the rest of the route r_2. If a route better than r_2 existed from J to K, it could be concatenated with r_1 to improve the route from I to K, contradicting our statement that r_1r_2 is optimal.



✓A subnet is represented by a graph, G = (V, E), where routers are the vertices and links are the edges. A tree is a connected undirected graph with no simple circuits

✓Consider a terminal of a network as the destination and the rest of the node are sources. Draw the optimal path from each source to the destination node. You will get a rooted tree where the destination node will be the root of the tree. Such tree is called sink tree. The sink tree of the network is not unique.

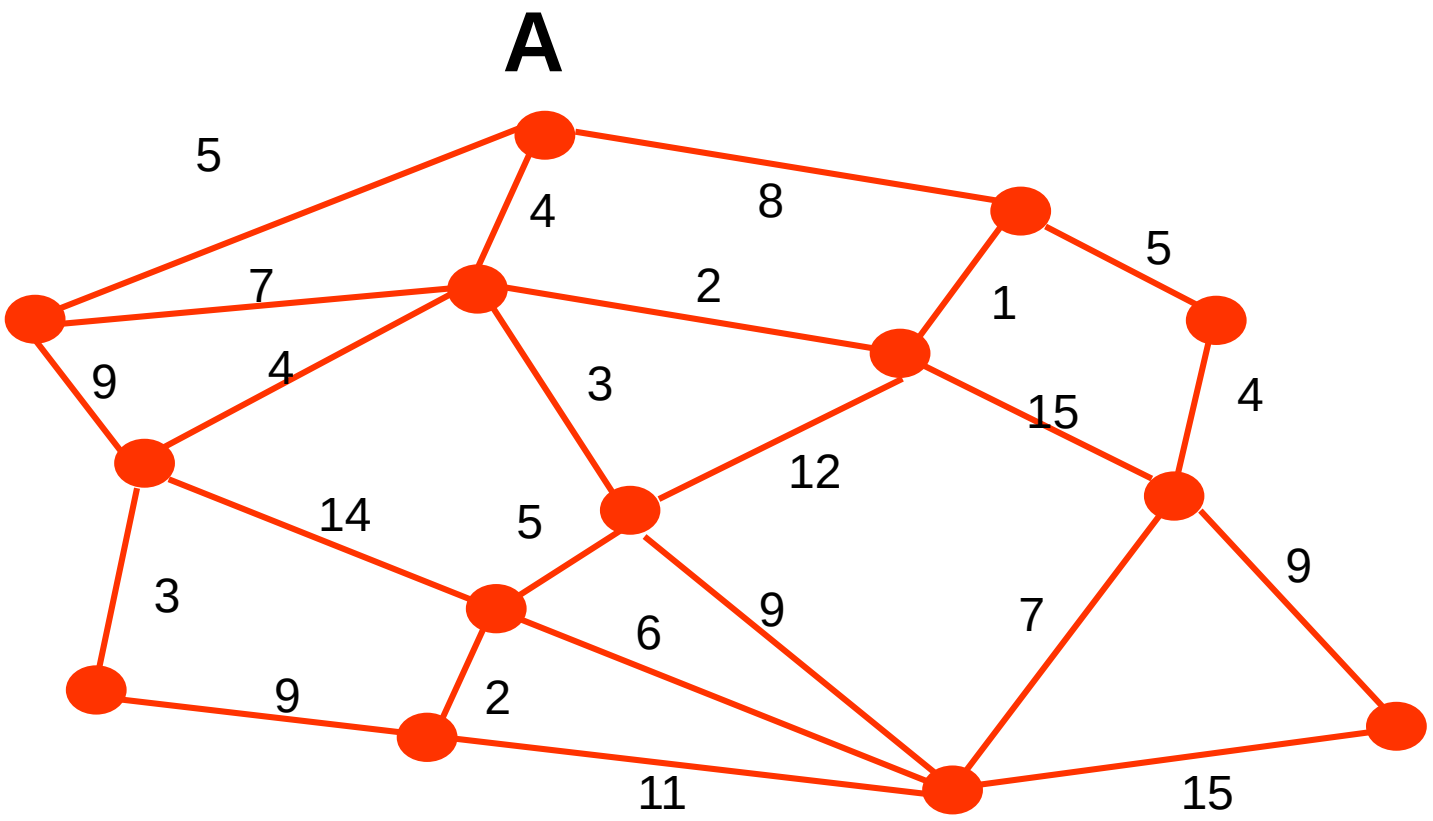


(a) A subnet. (b) A sink tree for router B

What is the drawback of such algorithm?

Exercise

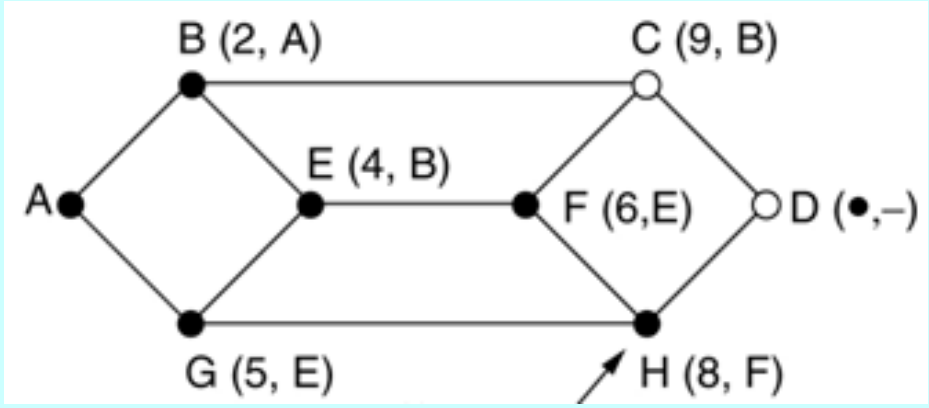
Draw the sink tree taking A as the root/destination.



Shortest Path Routing

✓The concept of shortest path deserves some explanation. One way of measuring path length is the *number of hops*. Using this metric, path *ABC* = path *ABE*.

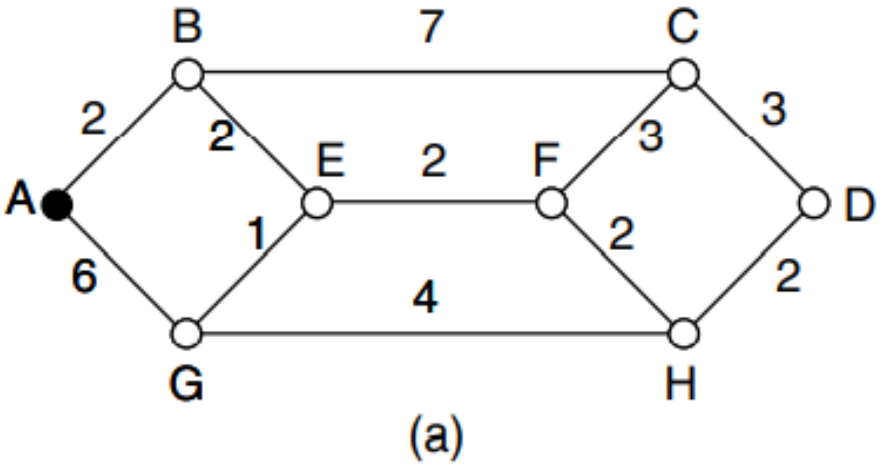
✓Another metric is the *geographic distance in Km*, in this case path path *ABC* > path *ABE*.



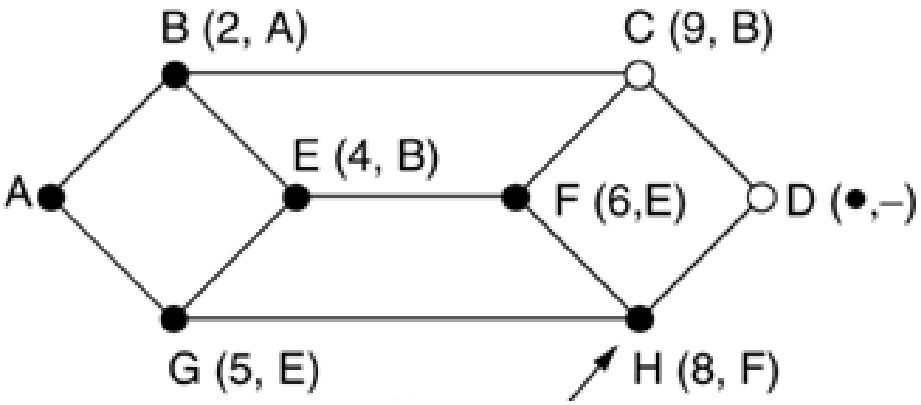
Usually path distance is function of:

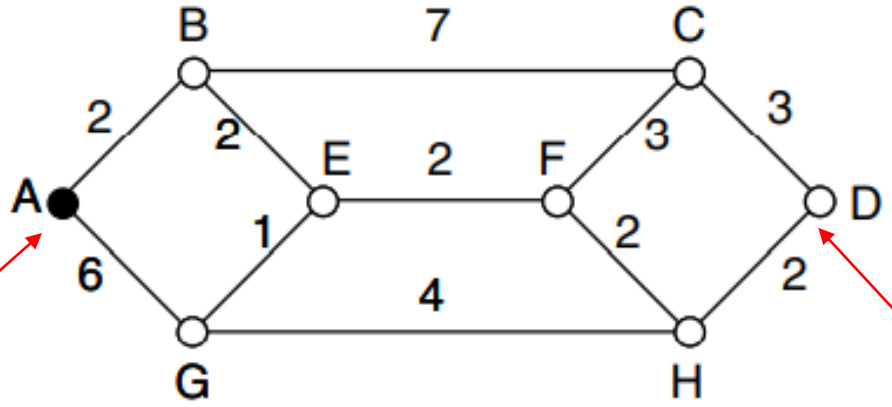
- propagation delay,
- BW/ link capacity in bps,
- average traffic load on the link,
- communication cost (in case of leased line),
- mean queue length of routers along the path (related to delay) etc.
- The combinations of above criteria the algorithm measures the shortest path.

✓Each link is weighted using above criteria (weighted graph). Each node is labeled (in parentheses) with its distance from the source node. Initially, no paths are known, so all nodes are labeled with infinity. As the algorithm proceeds and path are found, the labels may change, reflecting better paths.



(a)

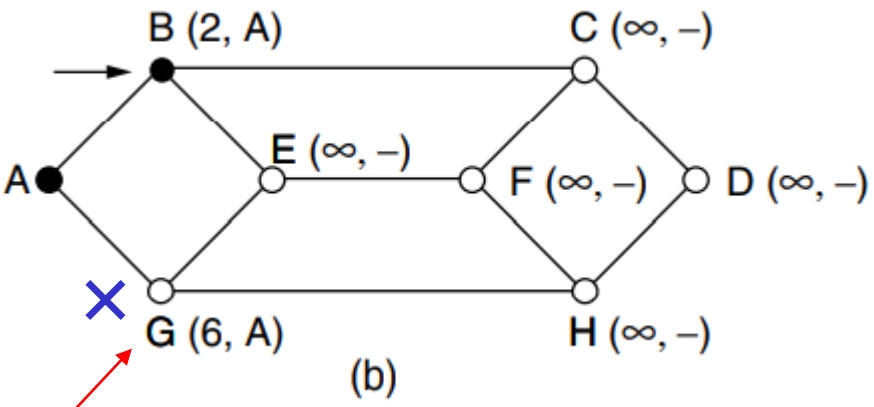




Source

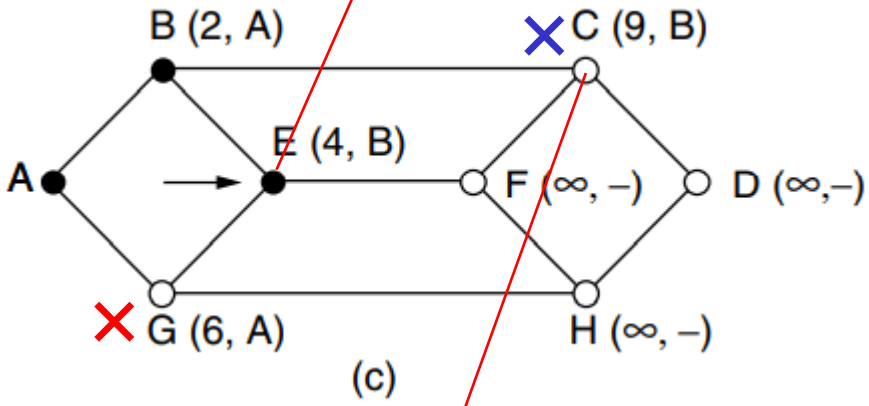
Destination

1st hop and minimum



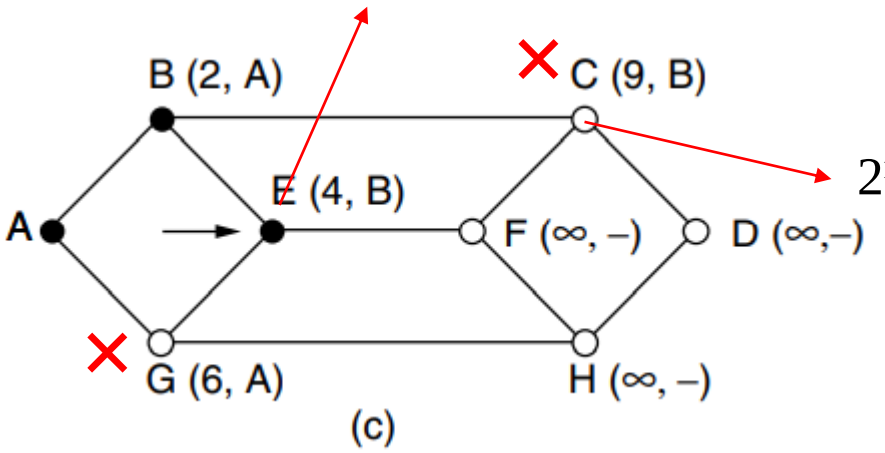
1st hop

2nd hop and minimum

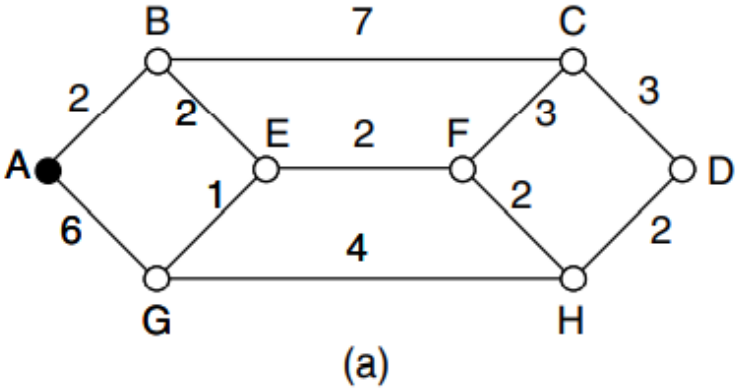


2nd hop

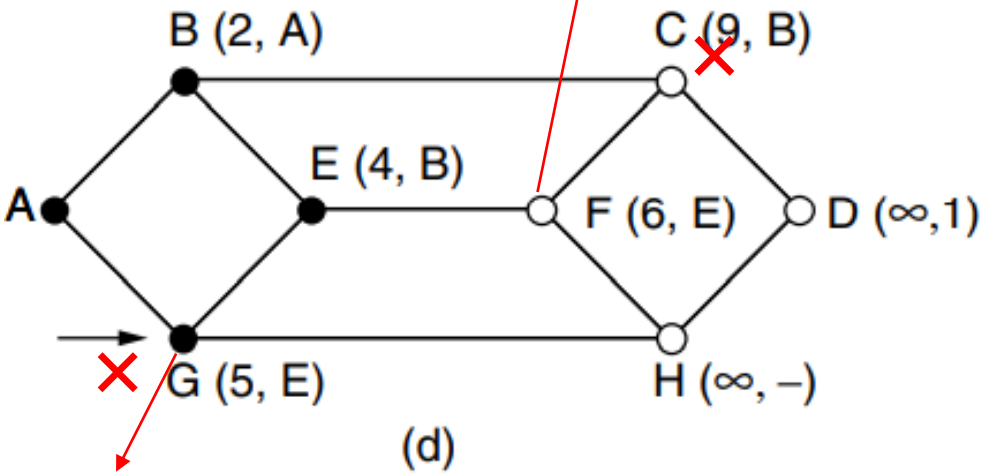
2nd hop and minimum

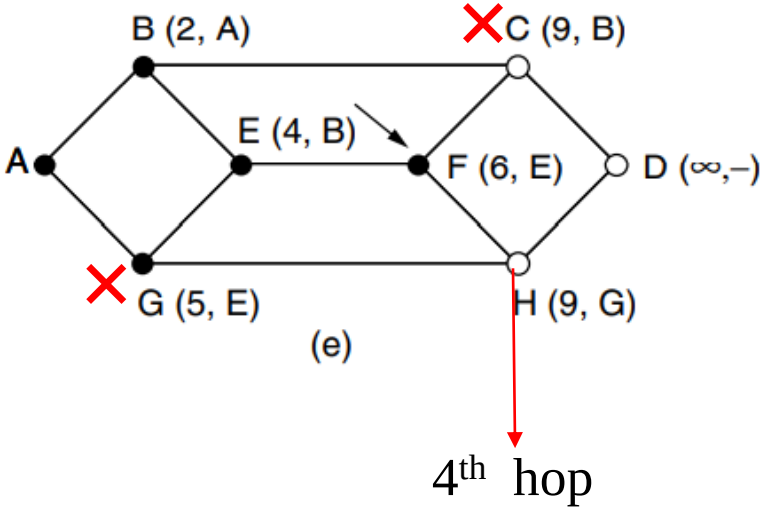


2nd hop



3rd hop



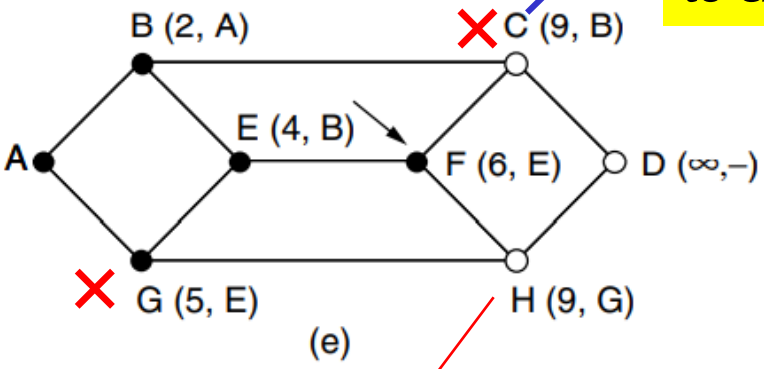


4th hop

3rd hop and minimum

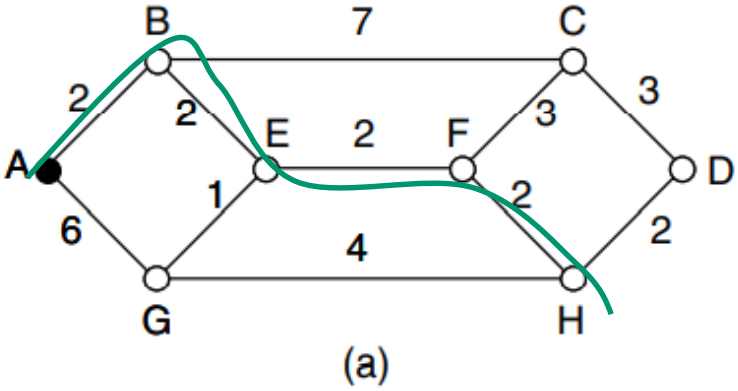
We arrived at this point G previously so it can not be on the best path so we have to choose alternate point i.e. point F.

We arrived at this point C previously so it can not be on the best path so we have to choose alternate point i.e. point H.

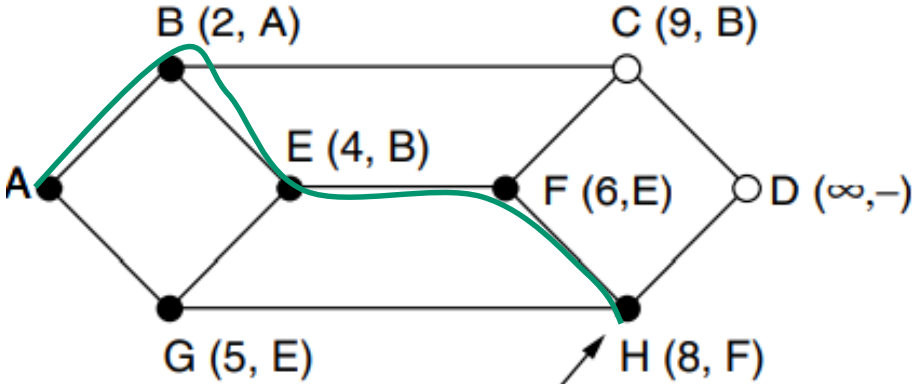


(e)

4th hop

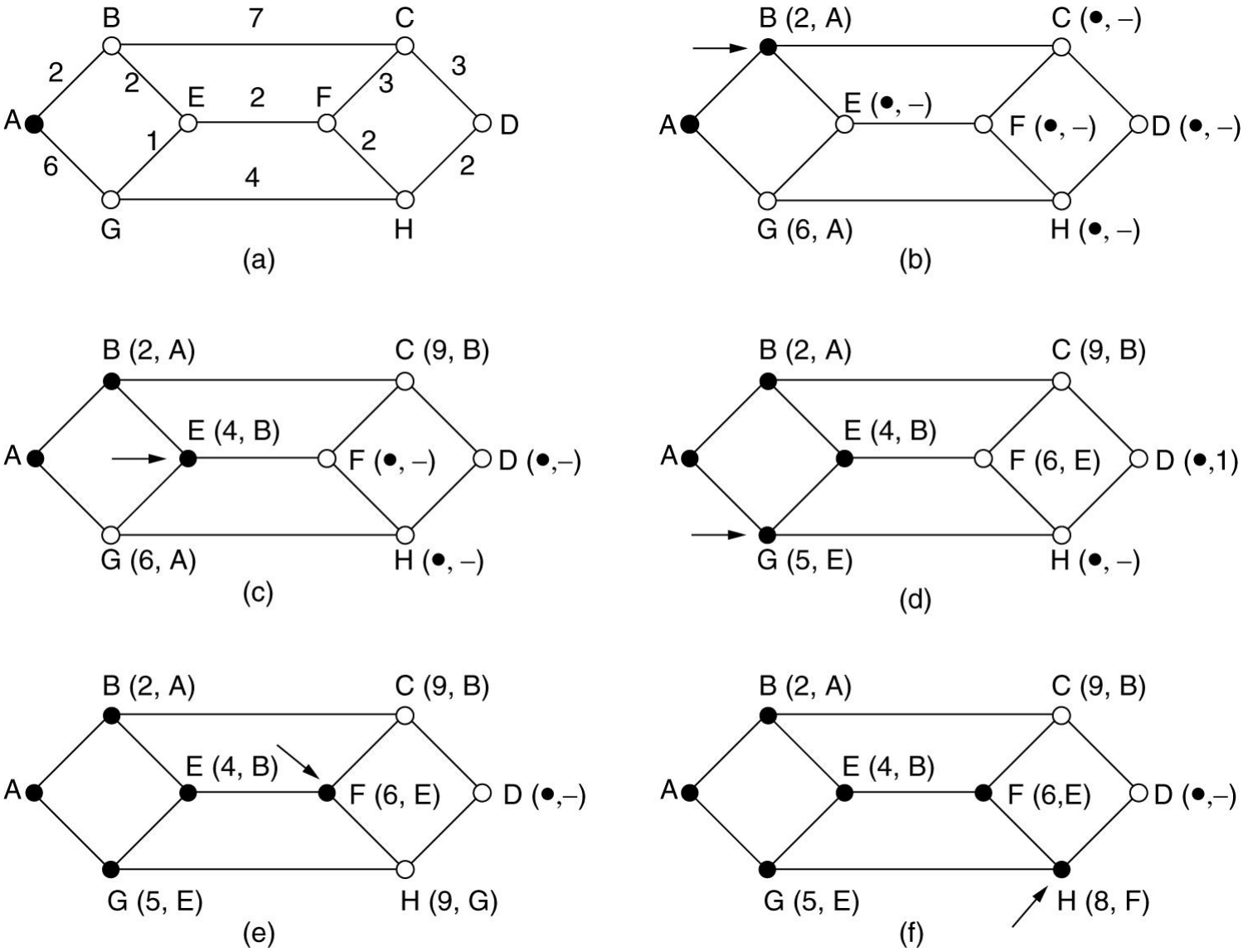


(a)



(f)

Shortest Path Routing Example

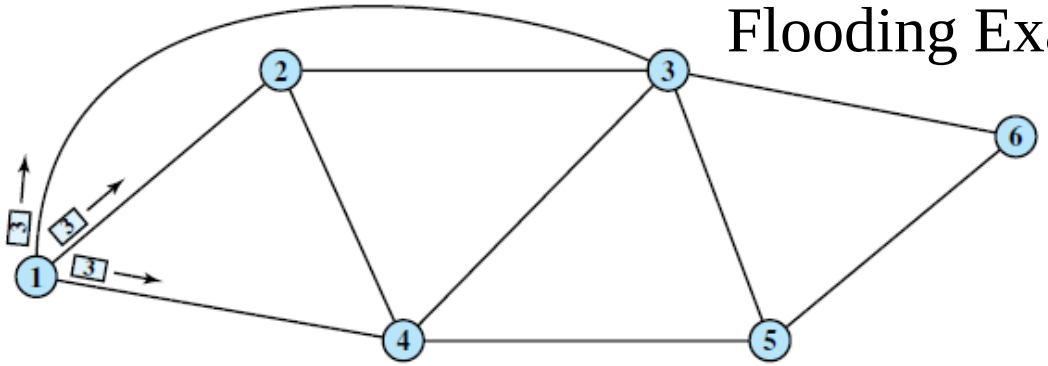


Flooding

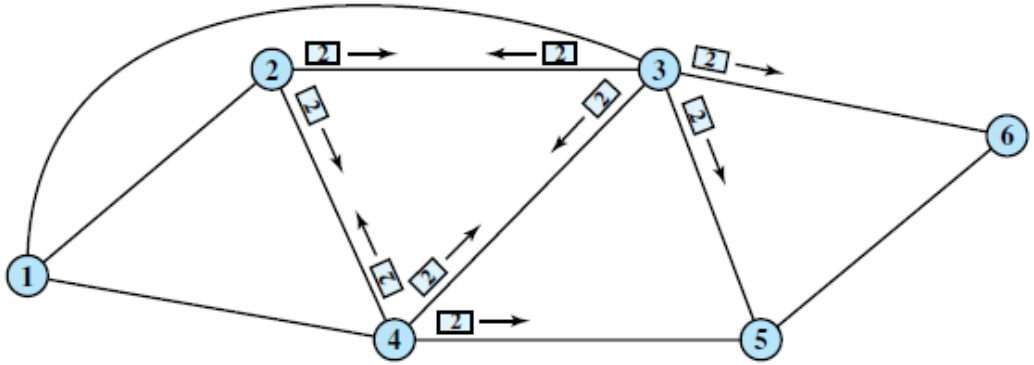
Flooding is a static algorithm in which every incoming packet is sent out on every outgoing line except the one it arrived on like HUB. Flooding obviously generates vast numbers of duplicating packets. Several remedial measures are taken against huge duplications of packets:

- Using hop counter in the header of each packet.
- An alternative technique for damming the flood is to keep track of which packets have been flooded, to avoid sending them out a second time.
- Using selective flooding algorithm where the routers do not send every incoming packet out on every line, only on those lines that are going approximately in the right direction.

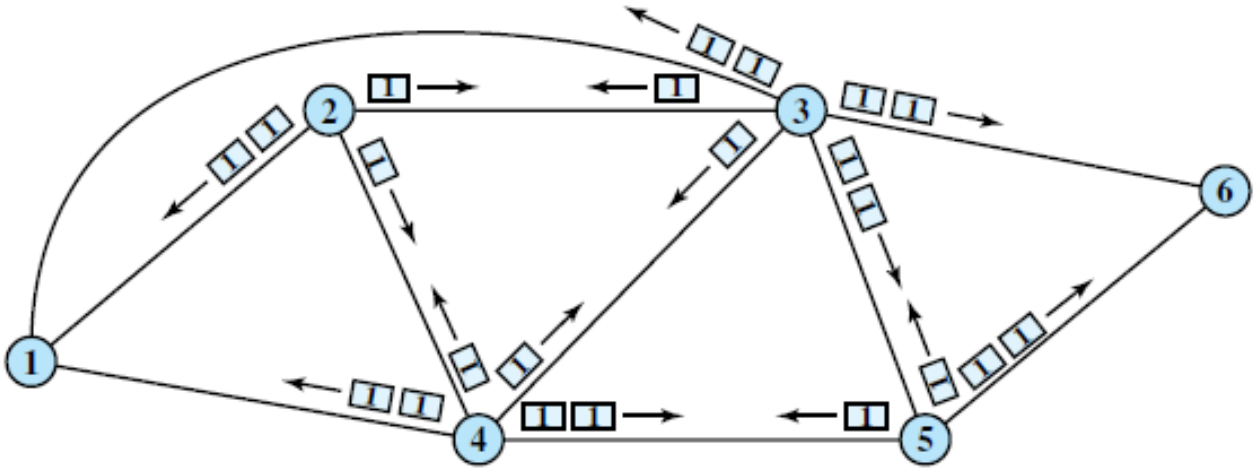
Flooding Example: hop count = 3



(a) First hop



(b) Second hop



(c) Third hop

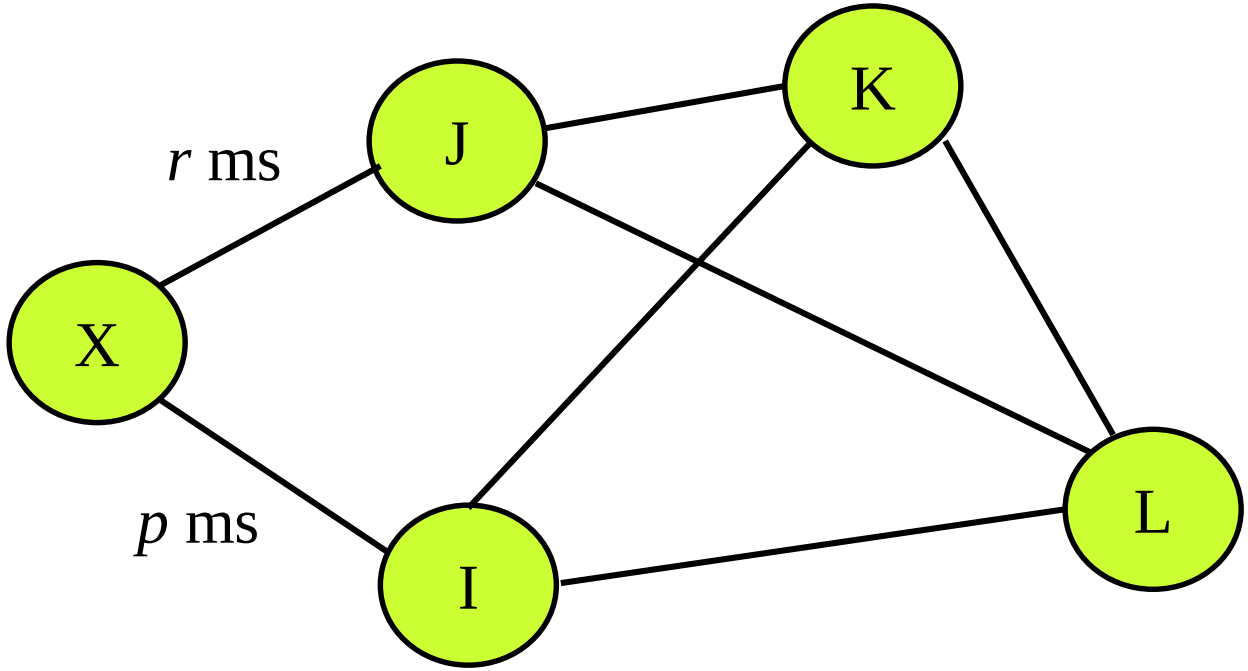
- Flooding is not practical in most applications, but it does have some uses. For example, in military applications, where a large number of routers may be blown at any instant, the tremendous robustness of flooding is highly desirable.

- In wireless networks, all messages are transmitted by a station can be received by all other stations within its radio range (because of unguided communications) which is in fact flooding.

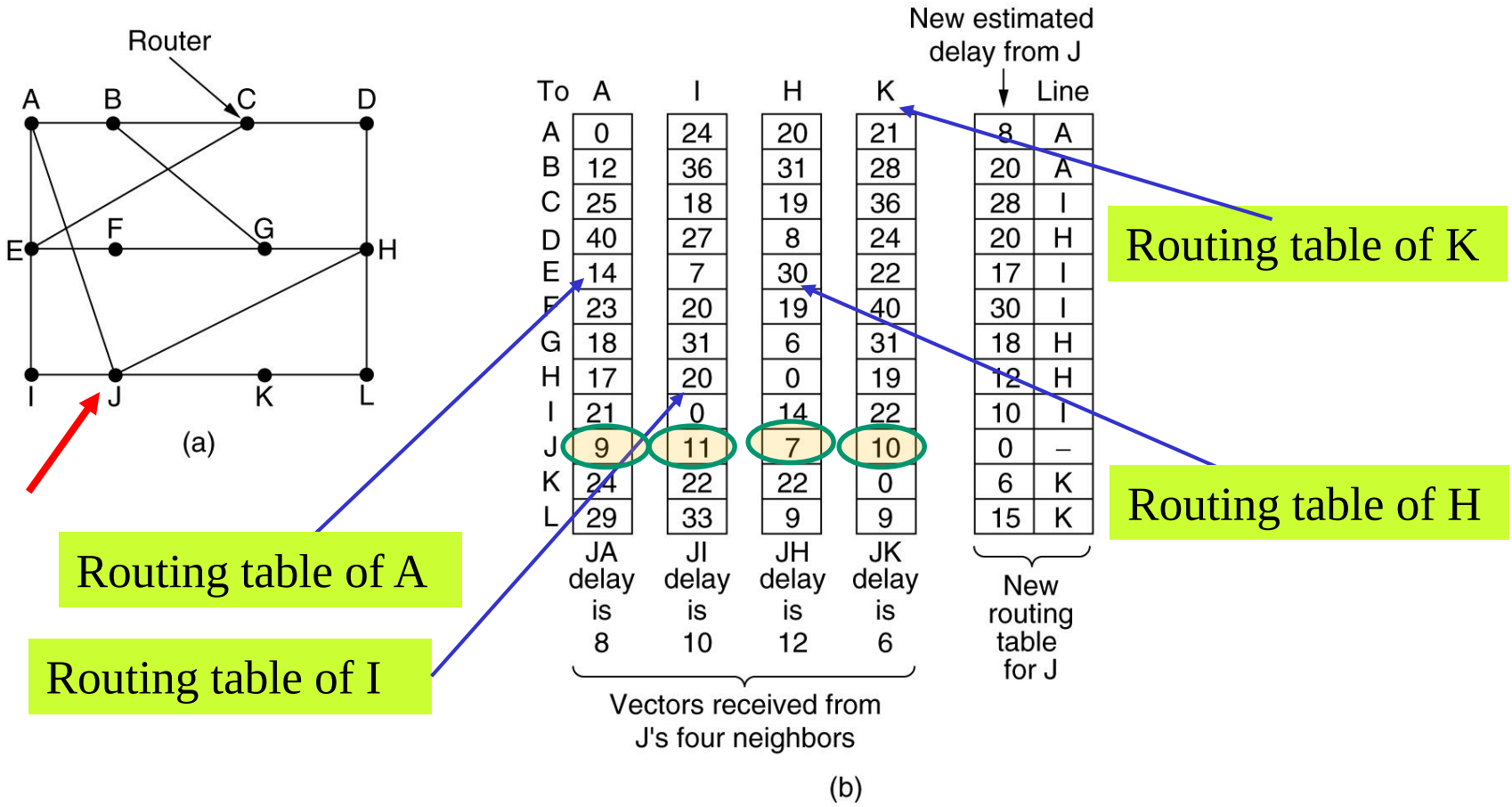
Distance Vector Routing

- ✓ **Two dynamic algorithms: distance vector routing** and **link state routing** are most popular.
- ✓ Distance vector routing algorithms operate by having each router maintain a table (vector) giving the best known distance to each destination and which line to use to get there. These tables are updated by exchanging information with the neighbors.
- ✓ The entry of the table contains two parts: the preferred outgoing line to use for that destination and an estimate of the time or distance to that destination.
- ✓ The metric used might be the number of hops, time delay in ms, total number of packets queued along the path, or some thing similar.

- ✓ Assume that delay is used as a metric and that router knows the delay to each of its neighbors. Once every *T* ms each router sends to each neighbor a list of its estimated delays to each destination. It also receives a similar lists from each neighbor.
- ✓ Let the delay from router *X* to *J* is *r* ms and the delay from *I* to *X* is *p* ms. Now the delay from I to J will be *r*+*p* ms. In this way each router updates its table. The routers can find its delay with neighbors sending echo packets.

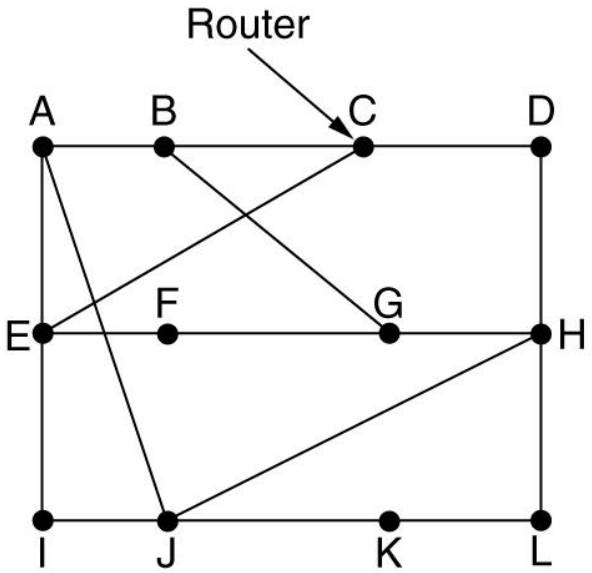


The updating process is illustrated in fig. below considering a subnet. Router J has four neighbors A, I, H, and K. The first four columns of fig.(b) shows the delay vectors received from the neighbors of router J.



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

Now assume that some delays are changed shown below the table. Now the new routing table of J will be:



(a)

Delay from J to G via A, JA+AG = 8 + 18 = 26ms

Delay from J to G via I, JI+IG = 10 + 31 = 41ms

Delay from J to G via H, JH+HG = 12 + 6 = 18ms

Delay from J to G via K, JK+KG = 6 + 31 = 37ms

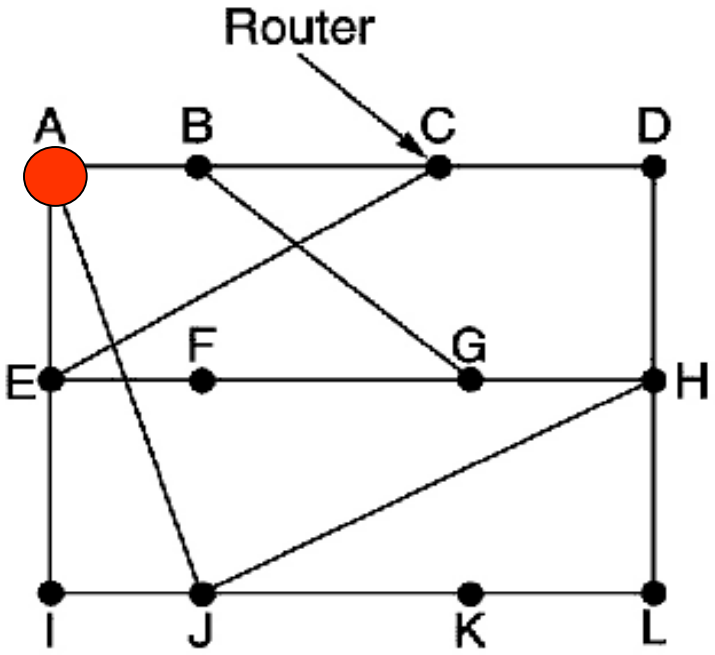
To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12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

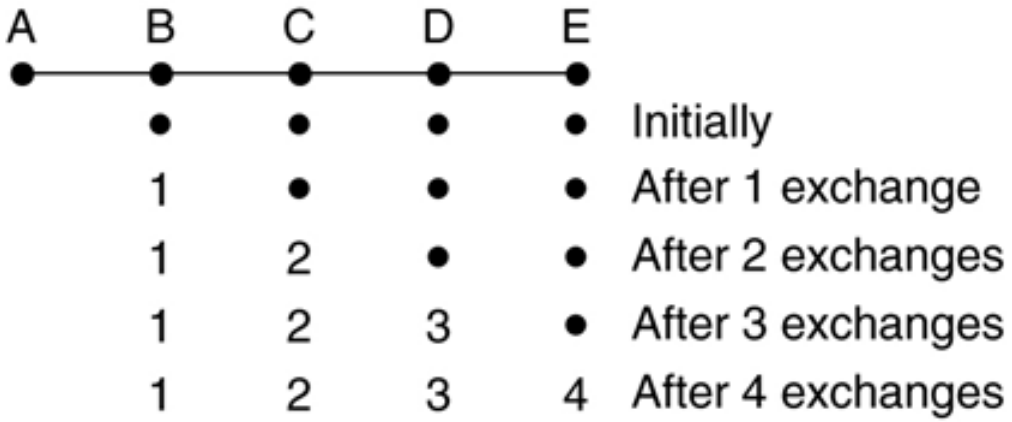
Vectors received from J's four neighbors

(b)

The count-to-infinity problem

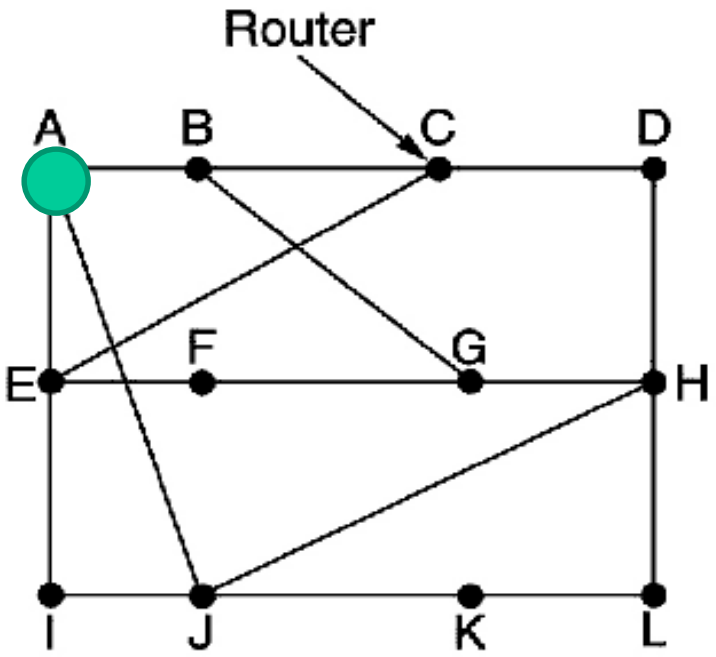
The distance vector routing react rapidly to good news, but leisure to bad news. To see how fast good news propagates, consider the five-node (linear) subnet of fig. below, where the delay metric is the number of hops. Suppose A is down initially and all the other routers know this. In other words, they have all recorded the delay to A as infinity.

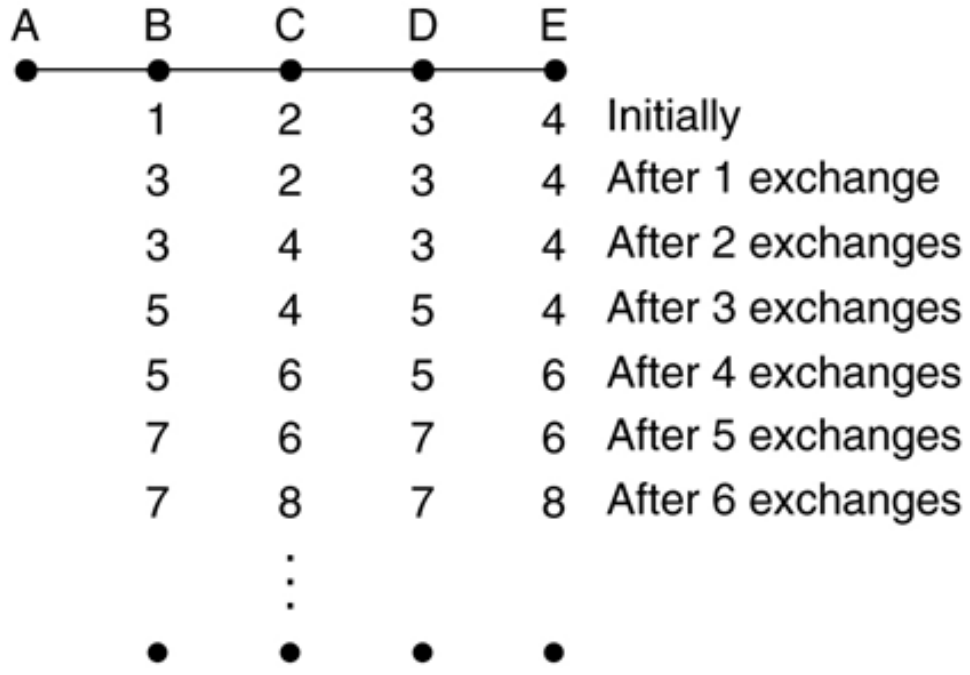




Good news

Now let us consider the situation of fig. below in which all the lines and routers are initially up. Routers B, C, D, and E have distances to A of 1, 2, 3, and 4, respectively. Suddenly A goes down, or alternatively, the line between A and B is cut, which is effectively the same thing from B's point of view.





Bad News

Link State Routing

Distance vector routing was used in the ARPANET until 1979, when it was replaced by **link state routing**. Two widely used algorithm are OSPF and IS-IS. The idea behind link state routing is fairly simple and can be stated as five parts.

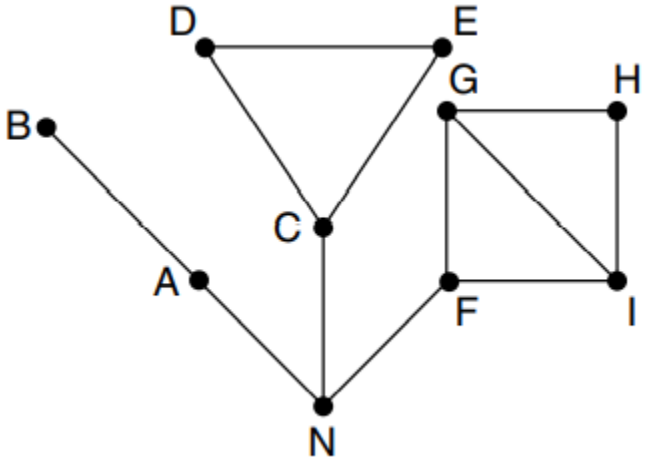
1. Discover its neighbors and learn their network addresses.
2. Set the distance or cost metric to each of its neighbors.
3. Construct a packet telling all it has just learned.
4. Send this packet to and receive packets from all other routers.
5. Compute the shortest path to every other router.

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 because when a distant router later hears that three routers are all connected to F, it is essential that it can determine whether all three mean the same F.



Setting Link Costs

- ✓ The link state routing algorithm requires each link to have a distance or cost metric for finding shortest paths.
- ✓ A common choice is to make the cost inversely proportional to the bandwidth of the link. For example, 1-Gbps Ethernet may have a cost of 1 and 100-Mbps Ethernet a cost of 10. This makes higher-capacity paths better choices.

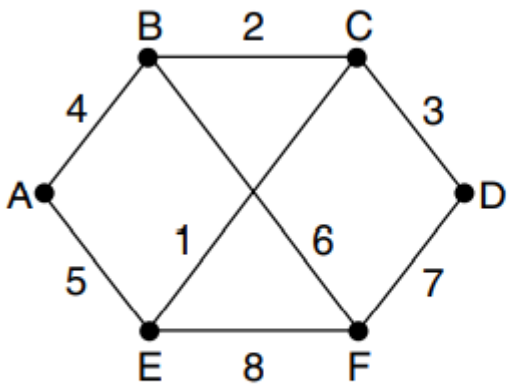
- ✓ If the network is geographically spread out, the delay of the links may be factored into the cost so that paths over shorter links are better choices. The most direct way to determine this delay is to send over the line a special ECHO packet that the other side is required to send back immediately.

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

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 starts with the identity of the sender, followed by a sequence number and age and a list of neighbors. The cost to each neighbor is also given.

- ✓ An example network is presented in Fig. (a) with costs shown as labels on the lines. The corresponding link state packets for all six routers are shown in Fig. (b).



(a)

Link State Packets

A	
Seq.	
Age	
B	4
E	5

B	
Seq.	
Age	
A	4
C	2
F	6

C	
Seq.	
Age	
B	2
D	3
E	1

D	
Seq.	
Age	
C	3
F	7

E	
Seq.	
Age	
A	5
C	1
F	8

F	
Seq.	
Age	
B	6
D	7
E	8

(b)

- ✓ Building the link state packets is easy. The hard part is determining when to build them. One possibility is to build them periodically, that is, 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.

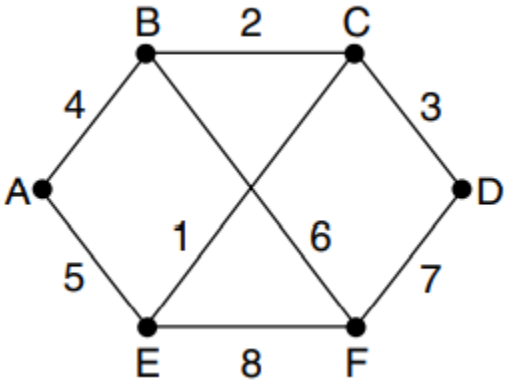
Distributing the Link State Packets

- ✓ All of the routers must get all of the link state packets quickly and reliably.
- ✓ When a link state packet comes in to 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.

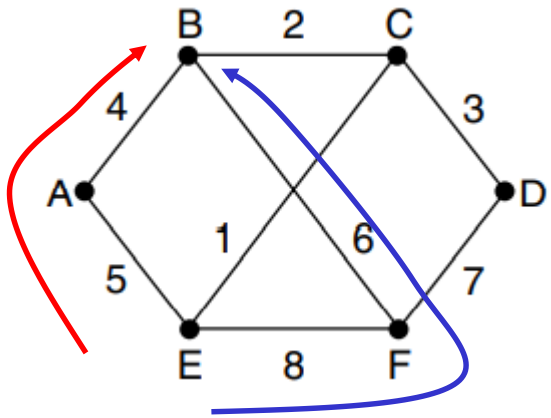
- ✓ To guard against errors on the links, all link state packets are acknowledged.

- ✓ The data structure used by router B for the network shown in Fig. below. The send flags mean that the packet must be sent on the indicated link. The acknowledgement flags mean that it must be acknowledged there.
- ✓ In Fig., the link state packet from A arrives directly, so it must be sent to C and F and acknowledged to A, as indicated by the flag bits. Similarly, the packet from F has to be forwarded to A and C and acknowledged to F.



			Send flags			ACK flags			
Source	Seq.	Age	A	C	F	A	C	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	

- ✓ However, the situation with the third packet, from E, is different. It arrives twice, once via EAB and once via EFB. Consequently, it has to be sent only to C but must be acknowledged to both A and F, as indicated by the bits.



			Send flags			ACK flags			
Source	Seq.	Age	A	C	F	A	C	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	

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.
- ✓ Compared to distance vector routing, link state routing requires more memory and computation.
- ✓ IS-IS (Intermediate System-Intermediate System) and OSPF (Open Shortest Path First) are the other main link state protocol