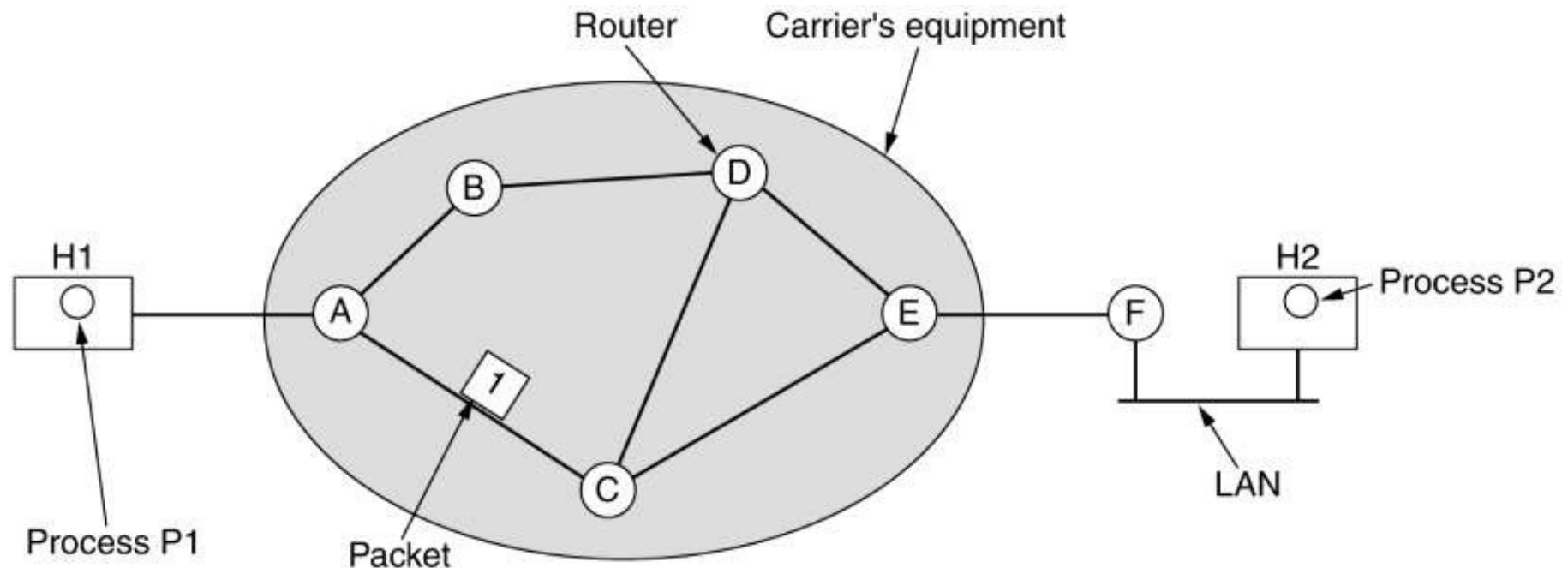


UNIT -III

Network Layer Design Issues

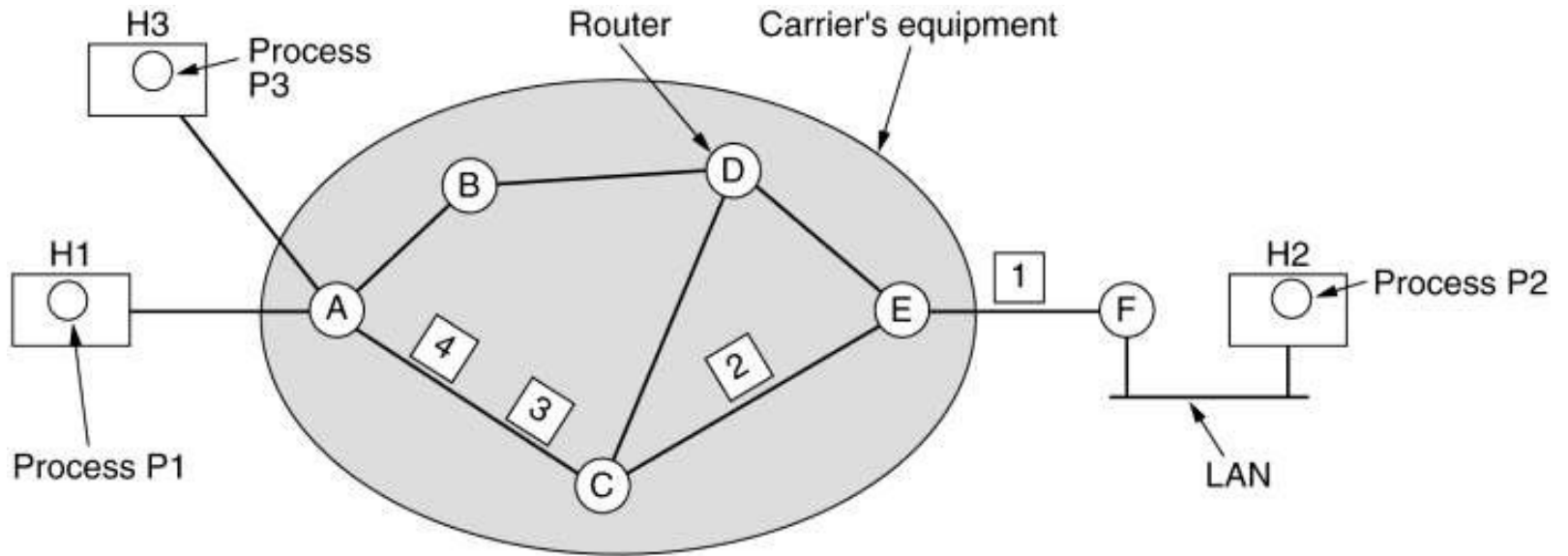
- Store-and-Forward Packet Switching
- Services Provided to the Transport Layer
- Implementation of Connectionless Service
- Implementation of Connection-Oriented Service
- Comparison of Virtual-Circuit and Datagram Subnets

Store-and-Forward Packet Switching



The environment of the network layer protocols.

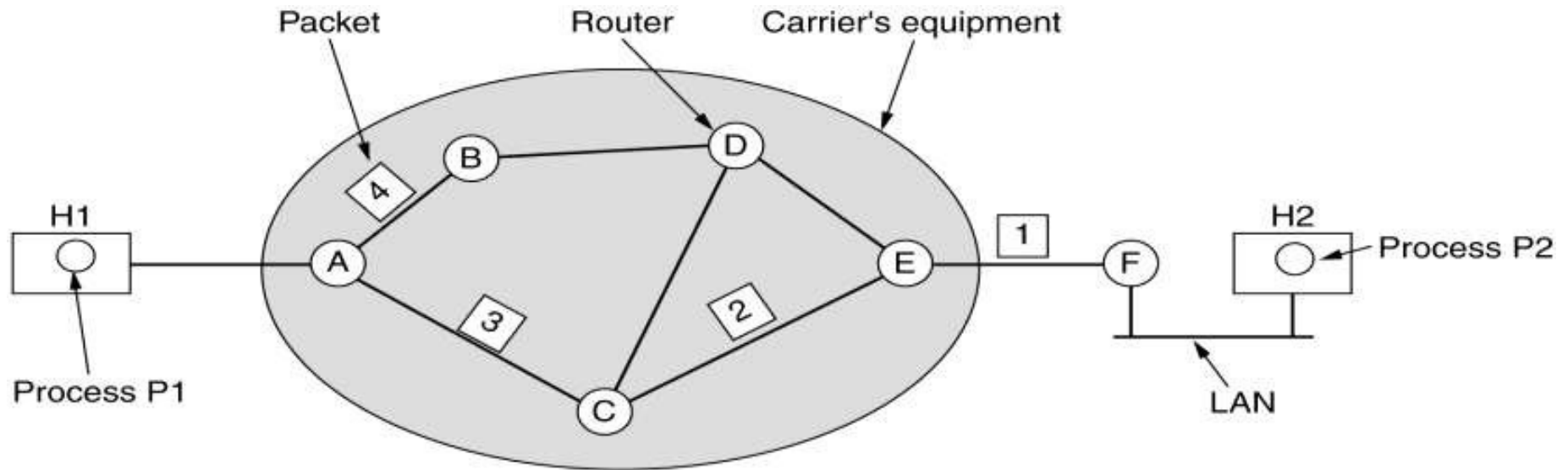
Implementation of Connection-Oriented Service



A's table		C's table		E's table	
H1	1	A	1	C	1
H3	1	A	2	C	2
In				Out	
		E	1	F	1
		E	2	F	2

Routing within a virtual-circuit subnet.

Implementation of Connectionless Service



A's table

initially	later
A	-
B	B
C	C
D	B
E	C
F	C

C's table

A	A
B	A
C	-
D	D
E	E
F	E

E's table

A	C
B	D
C	C
D	D
E	-
F	F

Dest. Line

Routing within a diagram subnet.

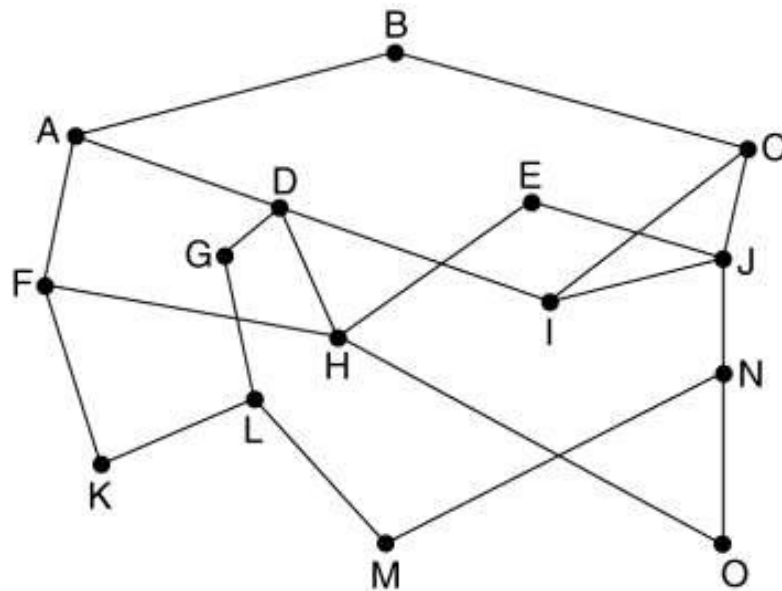
Comparison of Virtual-Circuit and Datagram Subnets

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

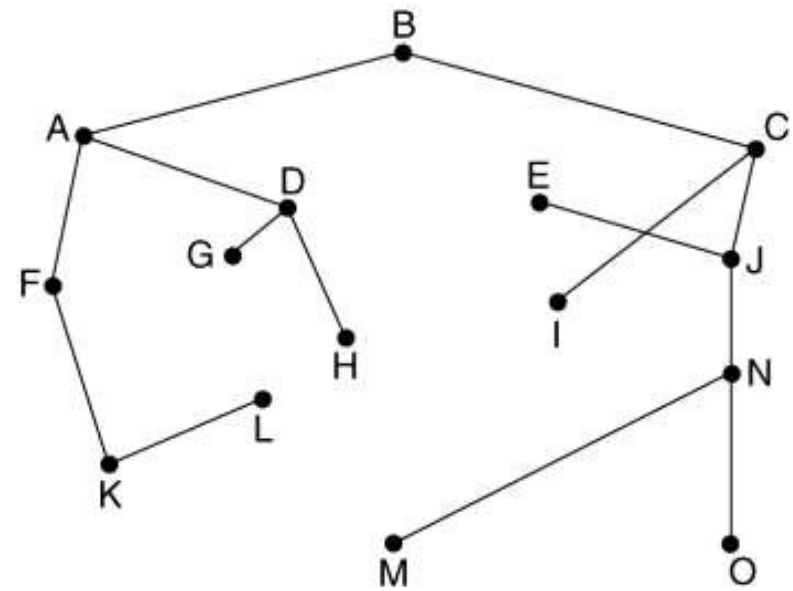
Routing Algorithms

- The Optimality Principle
- Shortest Path Routing
- Flooding
- Distance Vector Routing
- Link State Routing
- Hierarchical Routing

The Optimality Principle



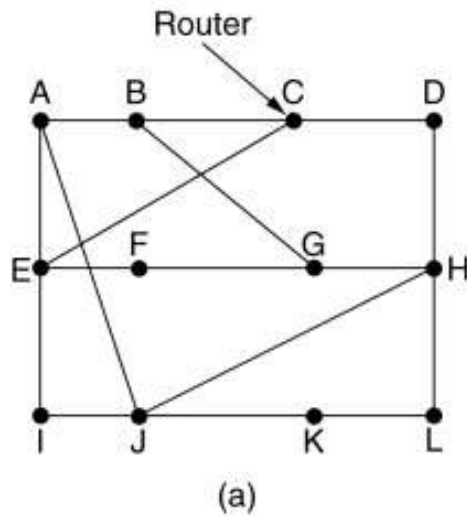
(a)



(b)

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

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

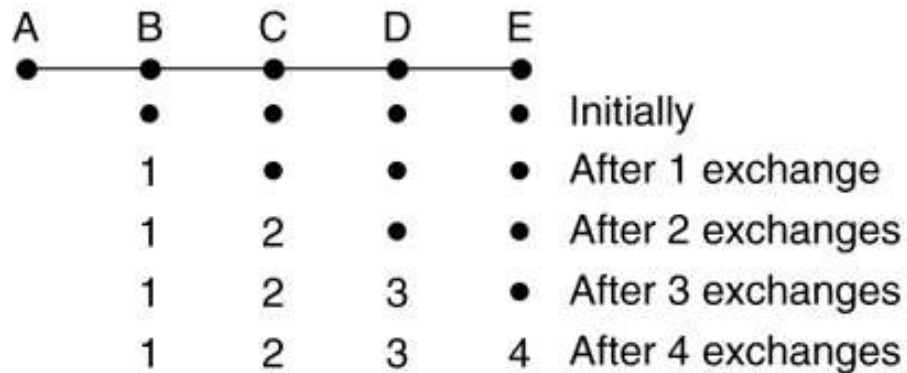
Vectors received from J's four neighbors

New routing table for J

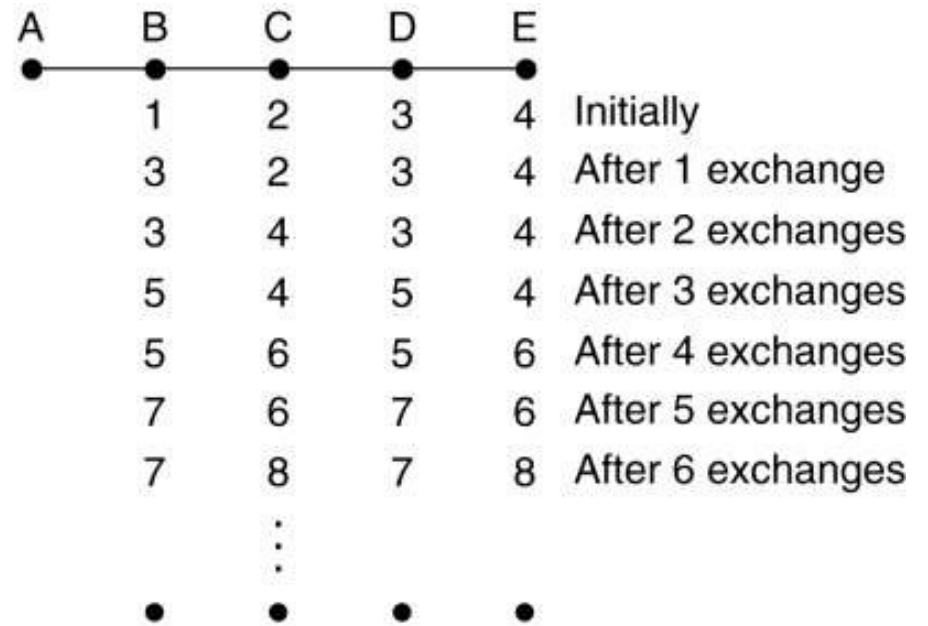
(b)

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

Distance Vector Routing (2)



(a)



(b)

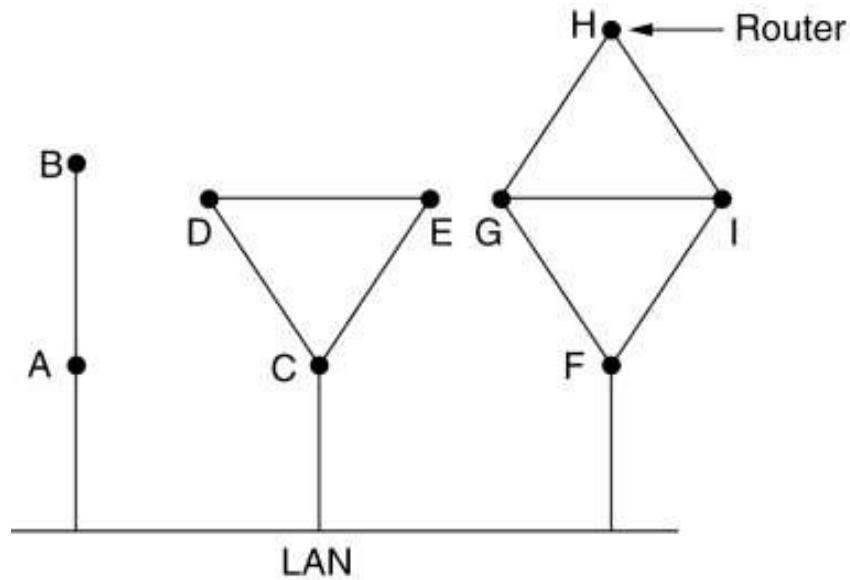
The count-to-infinity problem.

Link State Routing

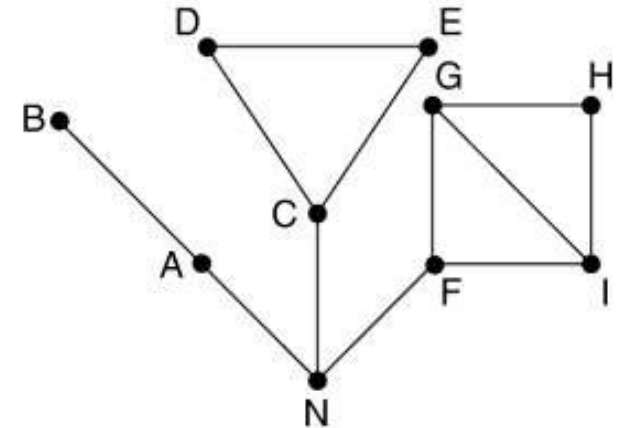
Each router must do the following:

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.

Learning about the Neighbors



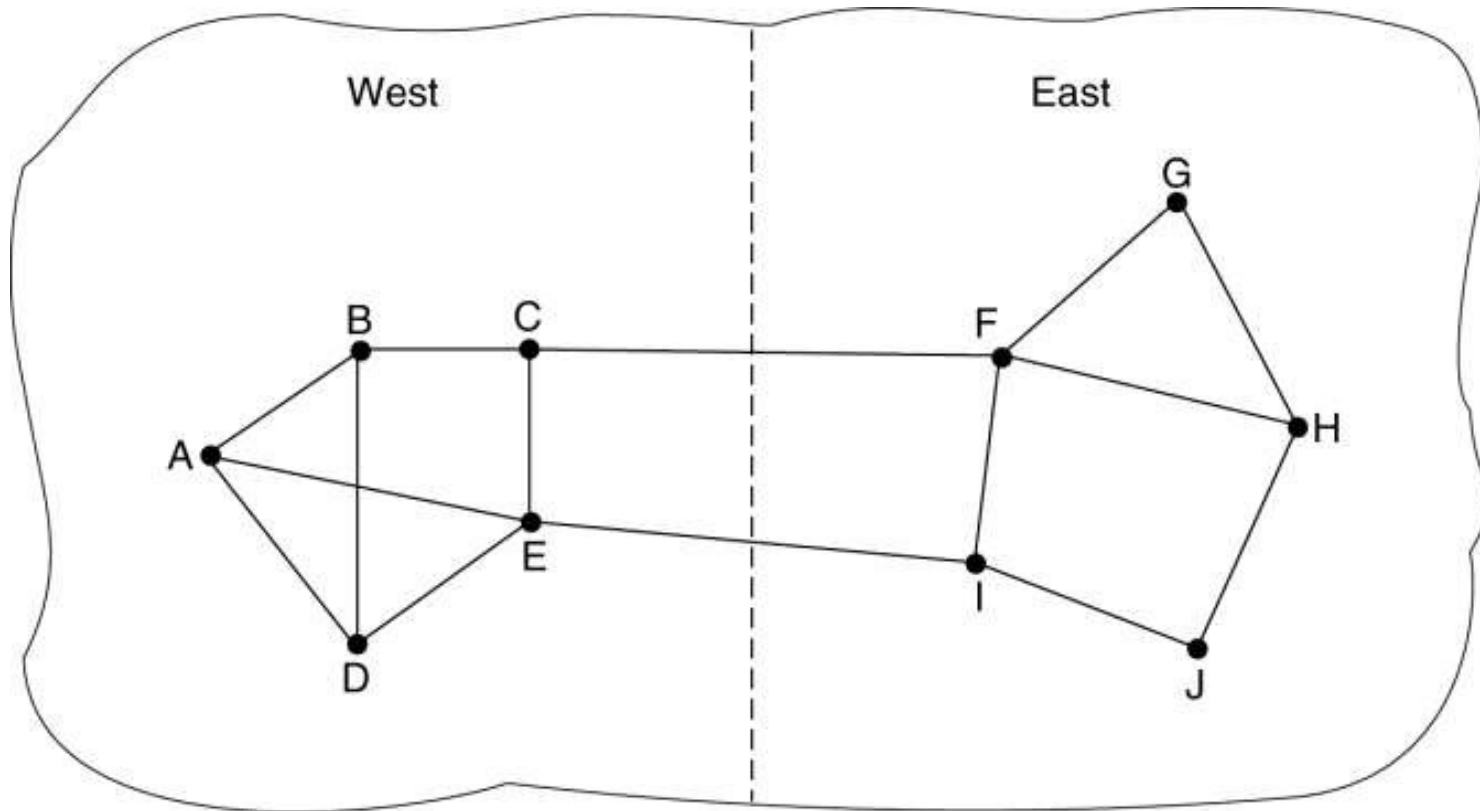
(a)



(b)

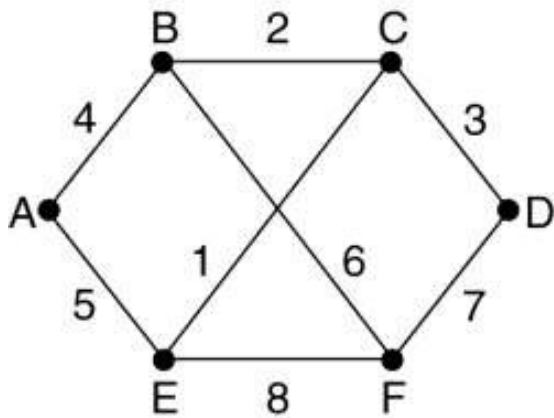
(a) Nine routers and a LAN. (b) A graph model of (a).

Measuring Line Cost



A subnet in which the East and West parts are connected by two lines.

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)

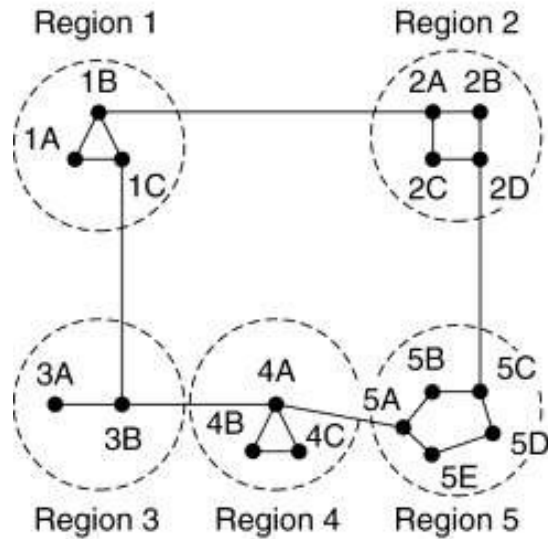
(a) A subnet. (b) The link state packets for this subnet.

Distributing the Link State Packets

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

The packet buffer for router B in the previous slide (Fig. 5-13).

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)

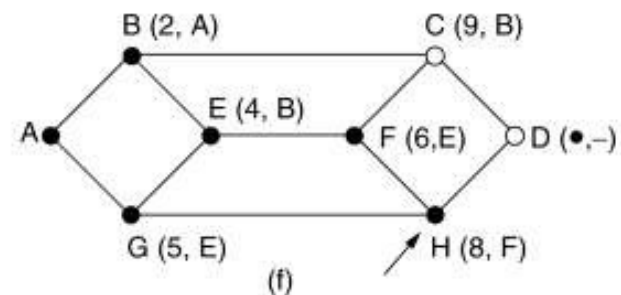
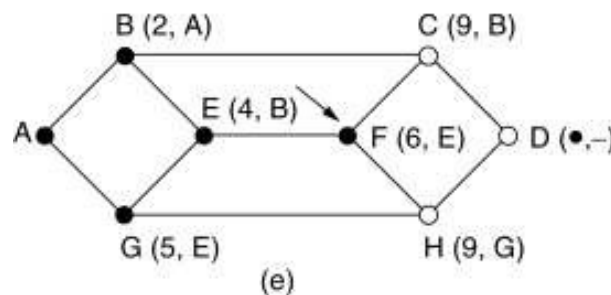
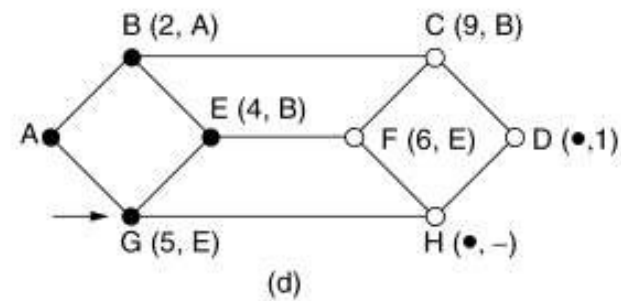
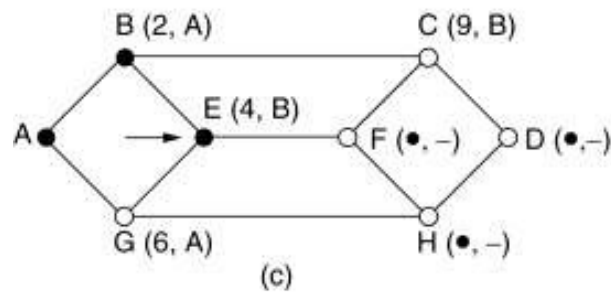
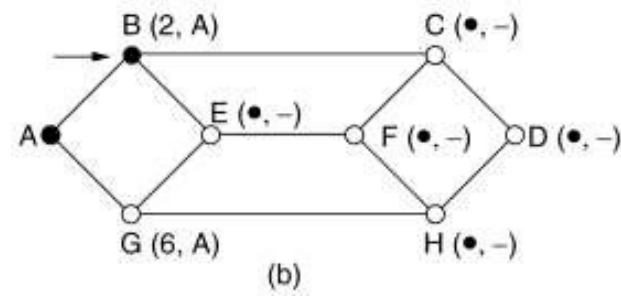
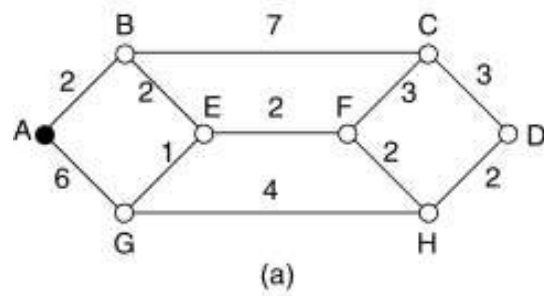
Hierarchical table for 1A

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

(c)

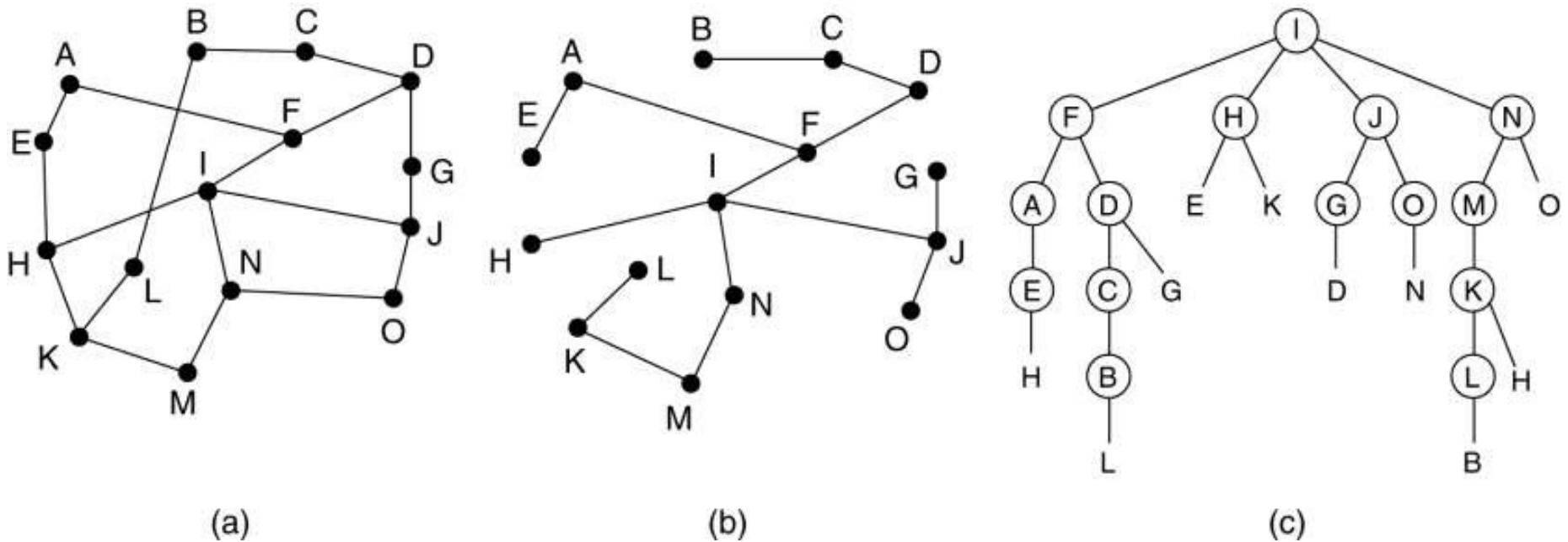
Hierarchical routing.

Shortest Path Routing



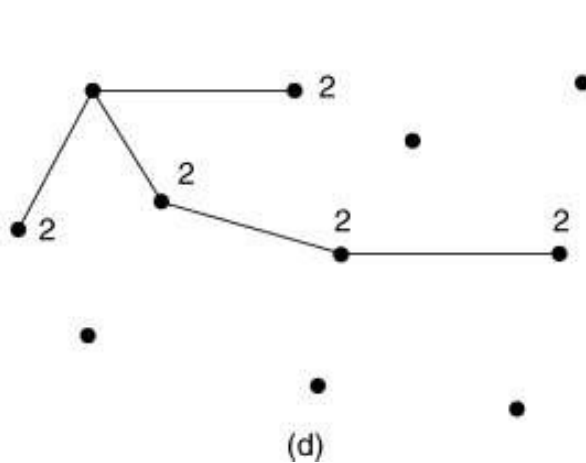
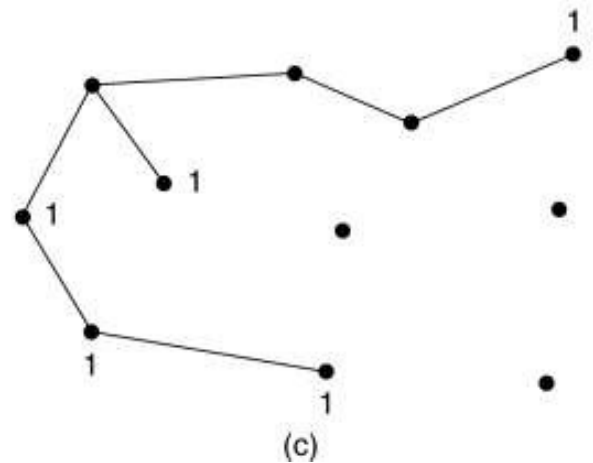
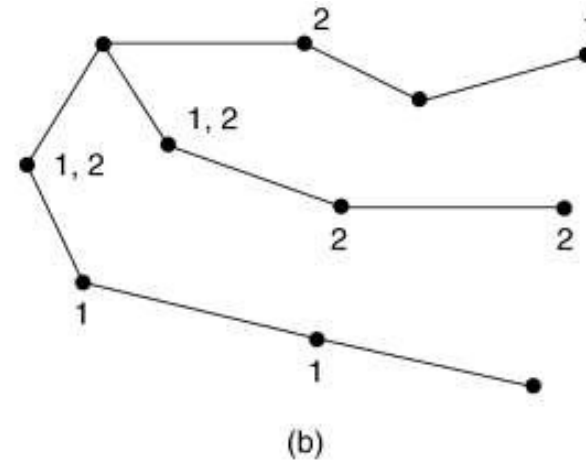
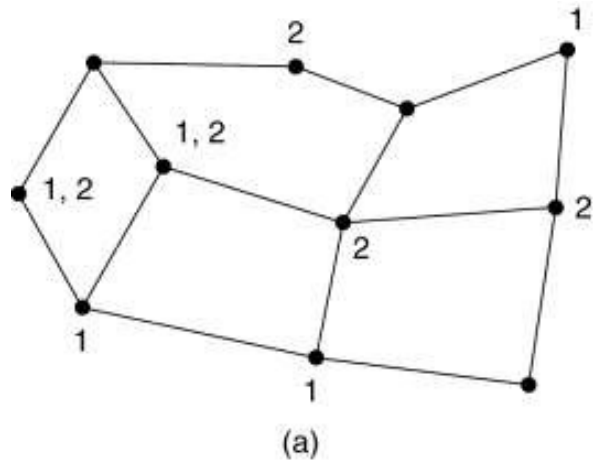
The first 5 steps used in computing the shortest path from A to D.
The arrows indicate the working node.

Broadcast Routing



Reverse path forwarding. (a) A subnet. (b) a Sink tree. (c) The tree built by reverse path forwarding.

Multicast Routing

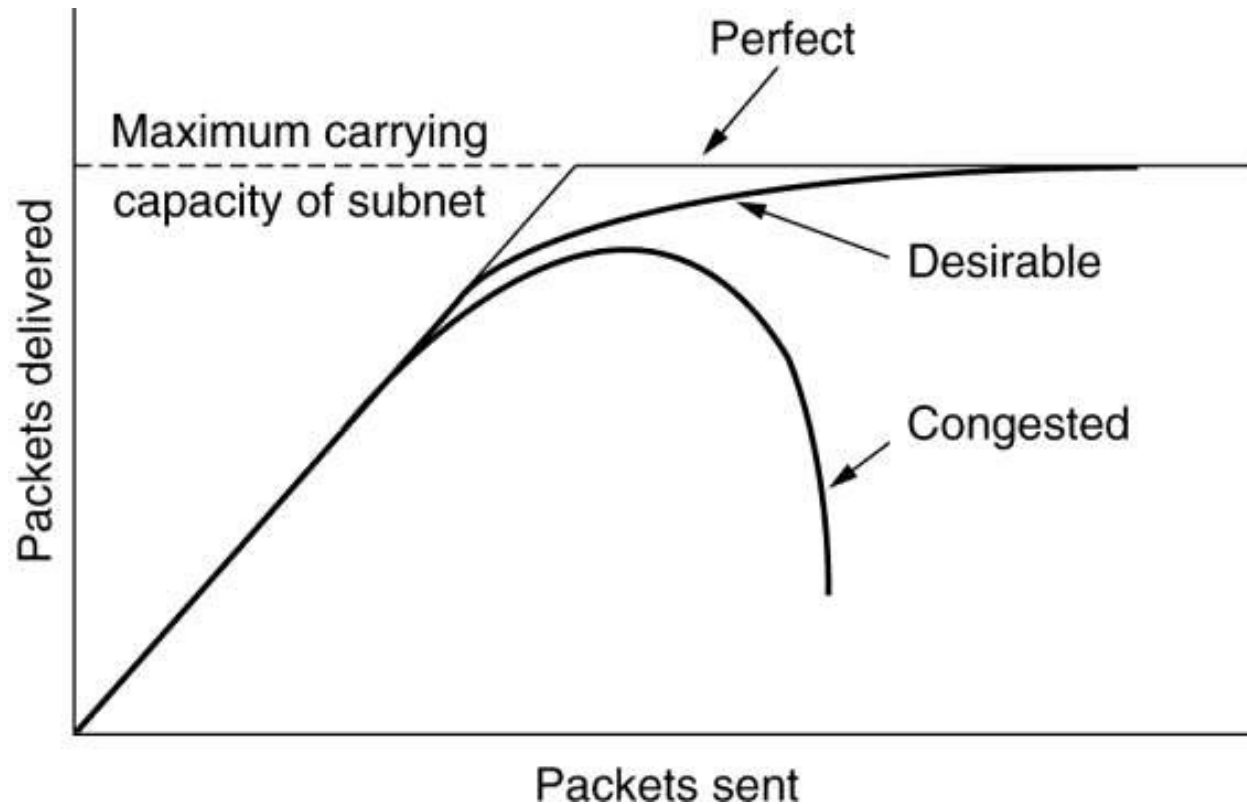


- (a) A network. (b) A spanning tree for the leftmost router.
(c) A multicast tree for group 1. (d) A multicast tree for group 2.

Congestion Control Algorithms

- General Principles of Congestion Control
- Congestion Prevention Policies
- Congestion Control in Virtual-Circuit Subnets
- Jitter Control

Congestion



When too much traffic is offered, congestion sets in and performance degrades sharply.

General Principles of Congestion Control

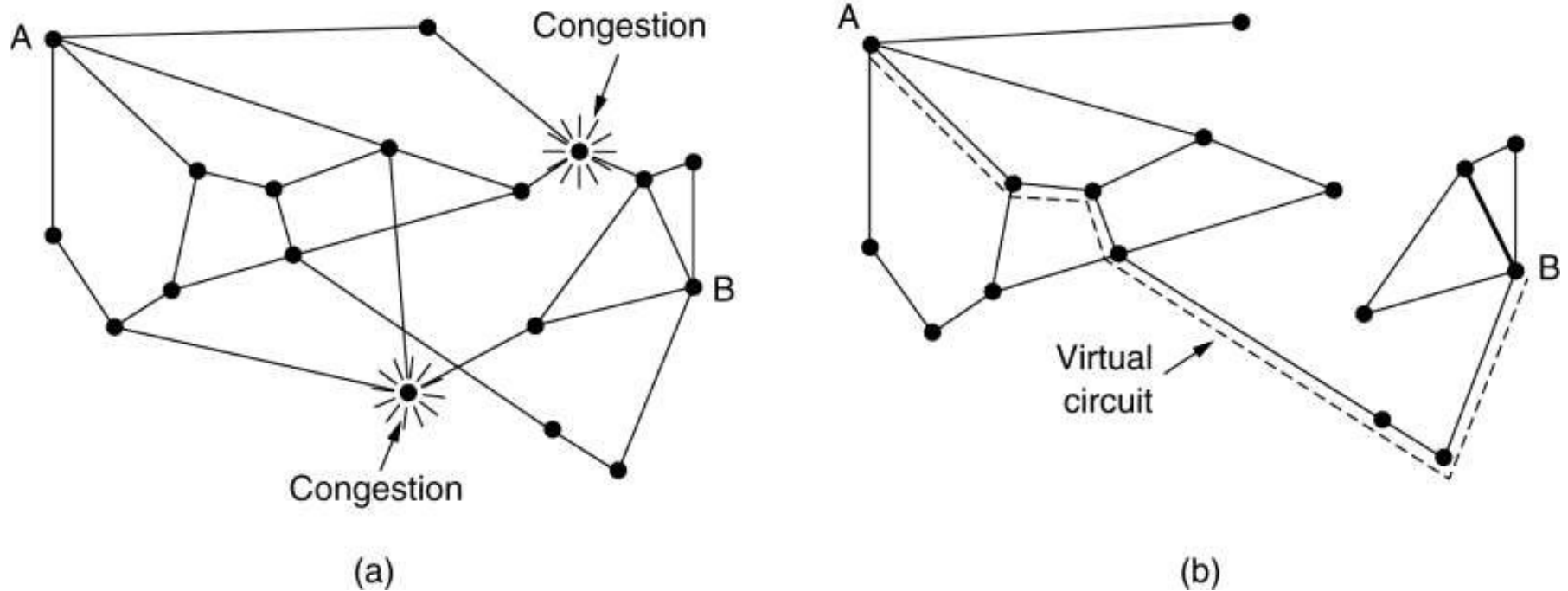
1. Monitor the system .
 - detect when and where congestion occurs.
2. Pass information to where action can be taken.
3. Adjust system operation to correct the problem.

Congestion Prevention Policies

Layer	Policies
Transport	<ul style="list-style-type: none">• Retransmission policy• Out-of-order caching policy• Acknowledgement policy• Flow control policy• Timeout determination
Network	<ul style="list-style-type: none">• Virtual circuits versus datagram inside the subnet• Packet queueing and service policy• Packet discard policy• Routing algorithm• Packet lifetime management
Data link	<ul style="list-style-type: none">• Retransmission policy• Out-of-order caching policy• Acknowledgement policy• Flow control policy

Policies that affect congestion.

Congestion Control in Virtual-Circuit Subnets

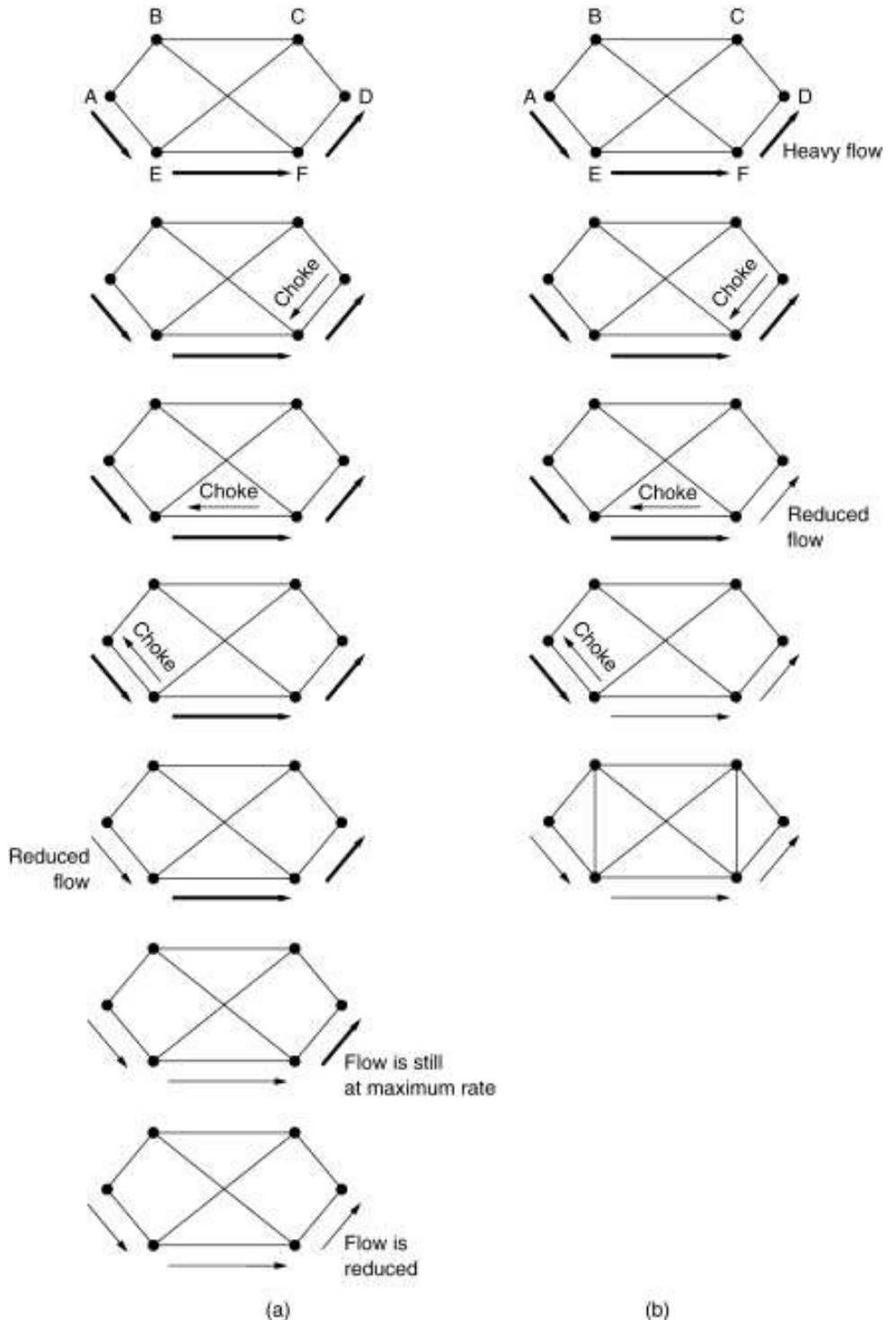


(a) A congested subnet. (b) A redrawn subnet, eliminates congestion and a virtual circuit from A to B.

Congestion Control in Datagram Subnets

- **The warning bit**
- **Choke packets**
- **Load shedding**
- **Random Early Detection**

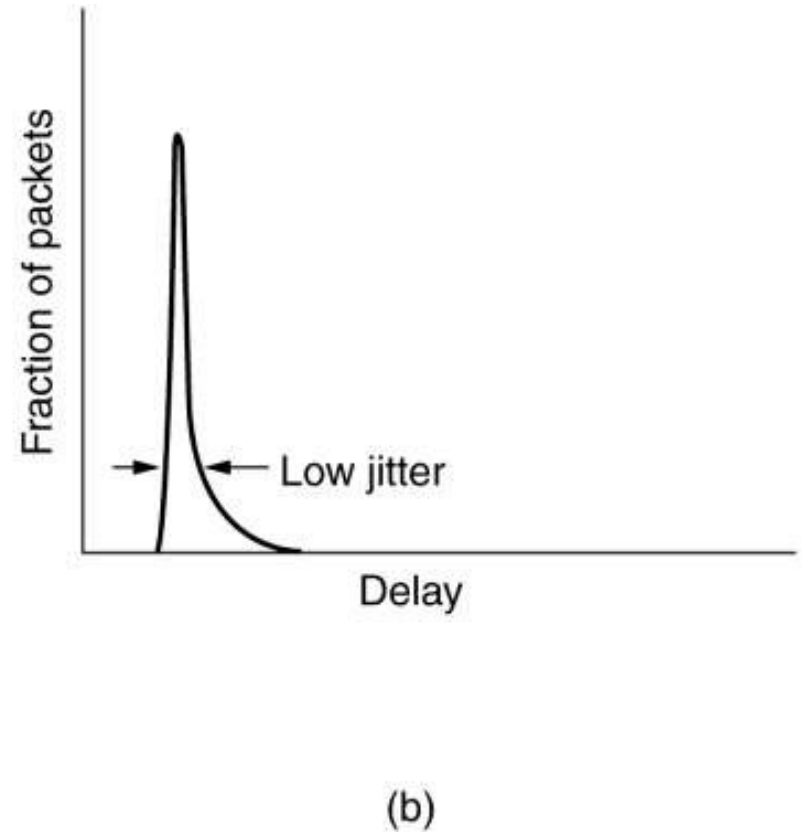
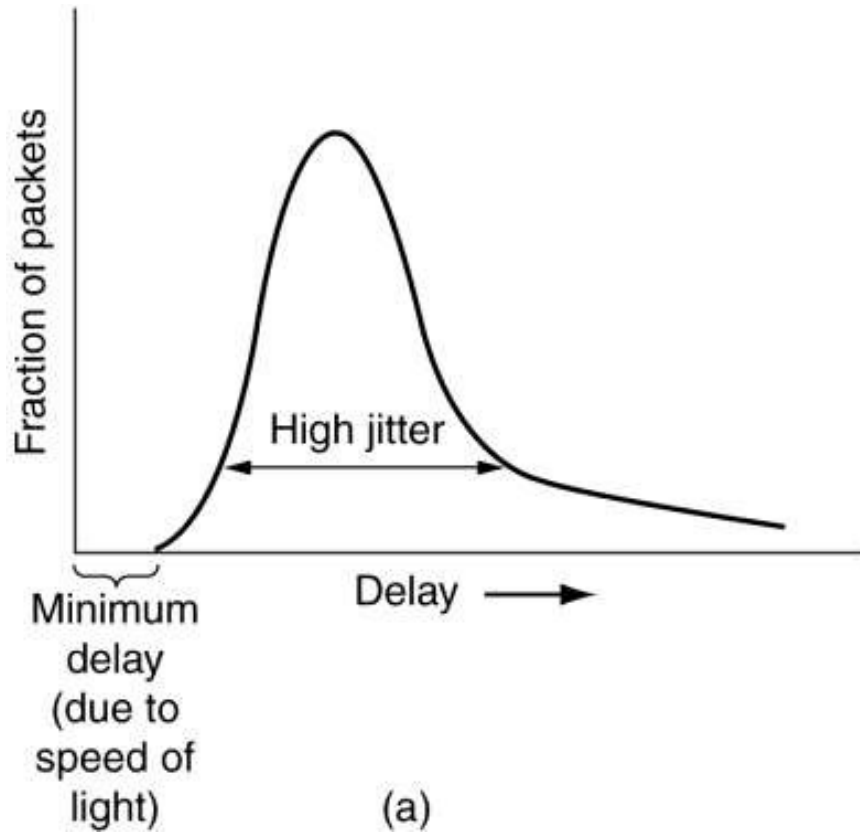
Hop-by-Hop Choke Packets



(a) A choke packet that affects only the source.

(b) A choke packet that affects each hop it passes through.

Jitter Control



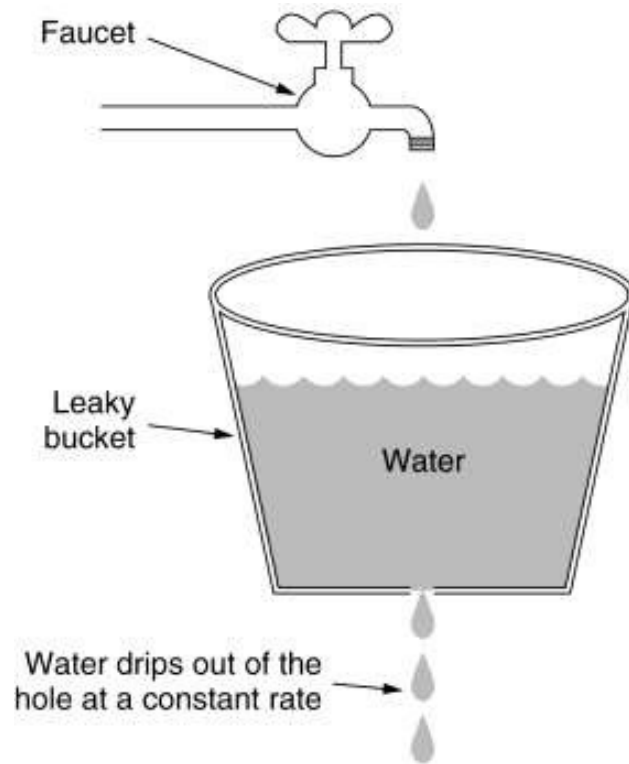
(a) High jitter.

(b) Low jitter.

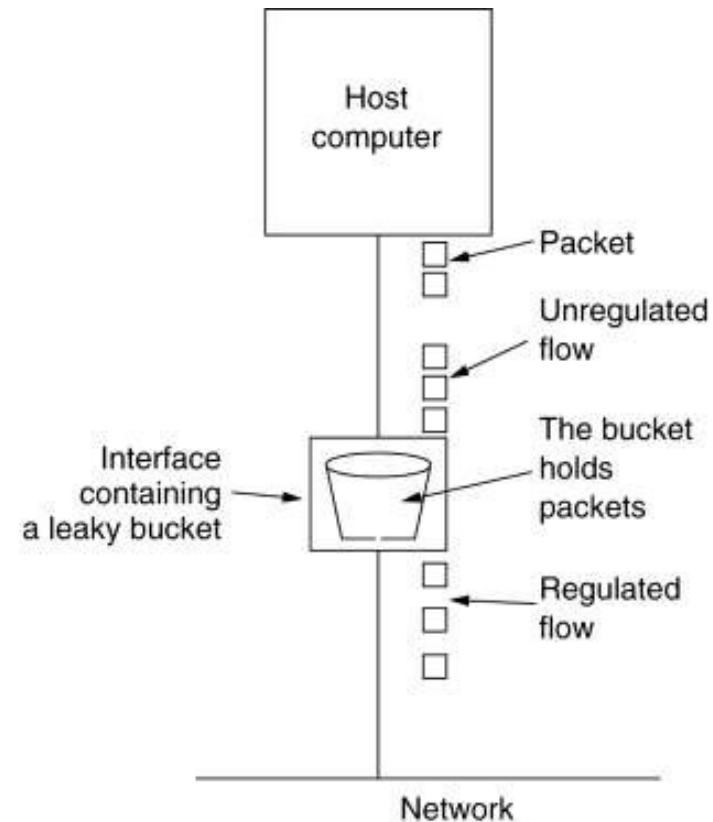
Congestion control Algorithms

- The Leaky Bucket Algorithm
- Token Bucket Algorithm

The Leaky Bucket Algorithm



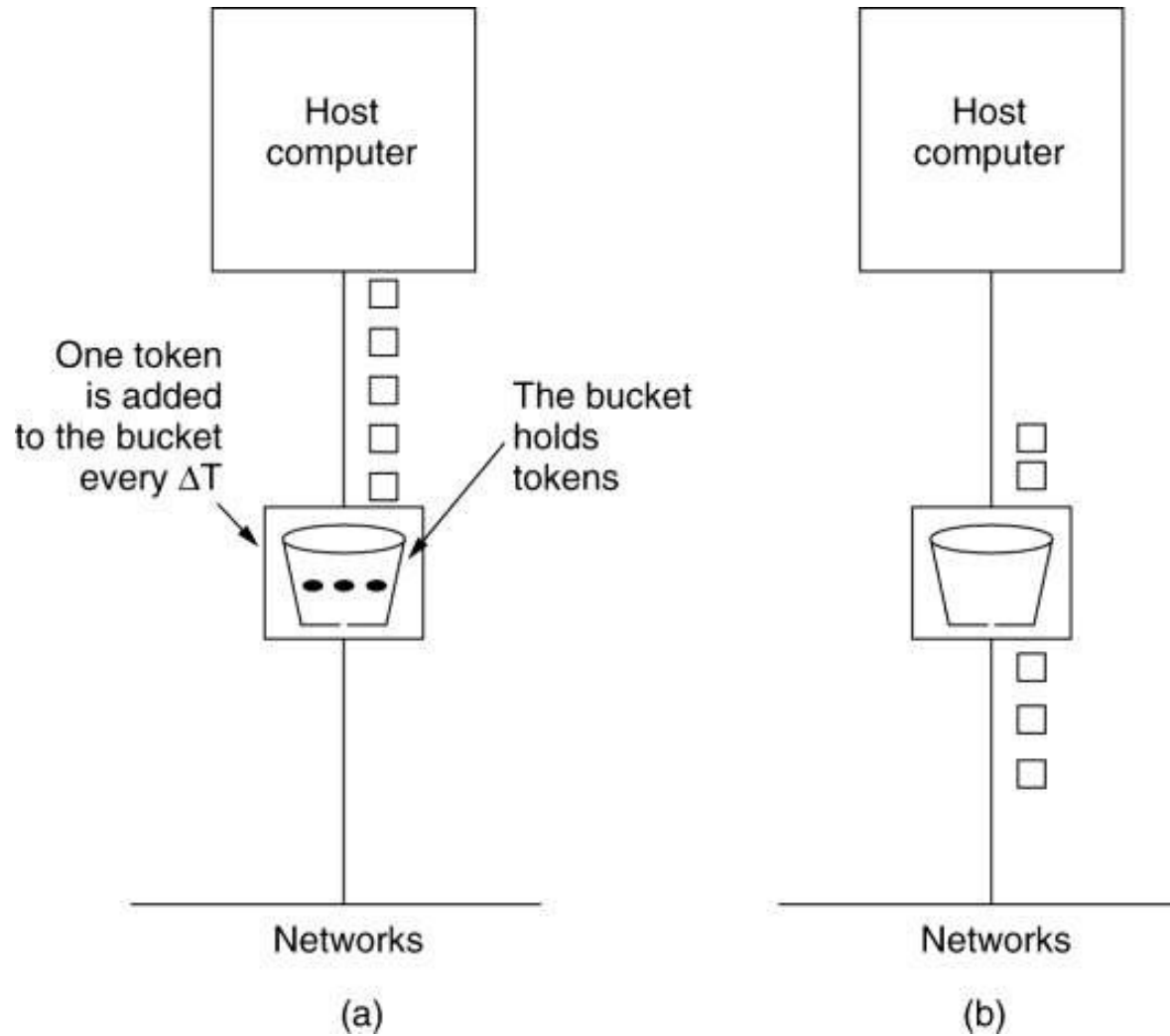
(a)



(b)

(a) A leaky bucket with water. (b) a leaky bucket with packets.

The Token Bucket Algorithm



(a) Before. (b) After.