COMPUTER NETWORKS
18CS46
UNIT 2

Dr. Minal Moharir

Computer Networks

Network Layer

Goal

end-to-end **transport of packets from source node all the way to destination node**

(compare with data link layer)

may require transfer through many sets of intermediate routers (multiple hops), in communication *subnets*

23.03.2020 17:11 2

Computer Networks

Network Layer

How

transport of packets is achieved through:

- **knowledge of subnets (set of all routers)**
- **choice of appropriate paths**
- **choice of paths also to take care of balancing the load on links / routers**

23.03.2020 17:11 3

Computer Networks
Network Layer

Design issues :

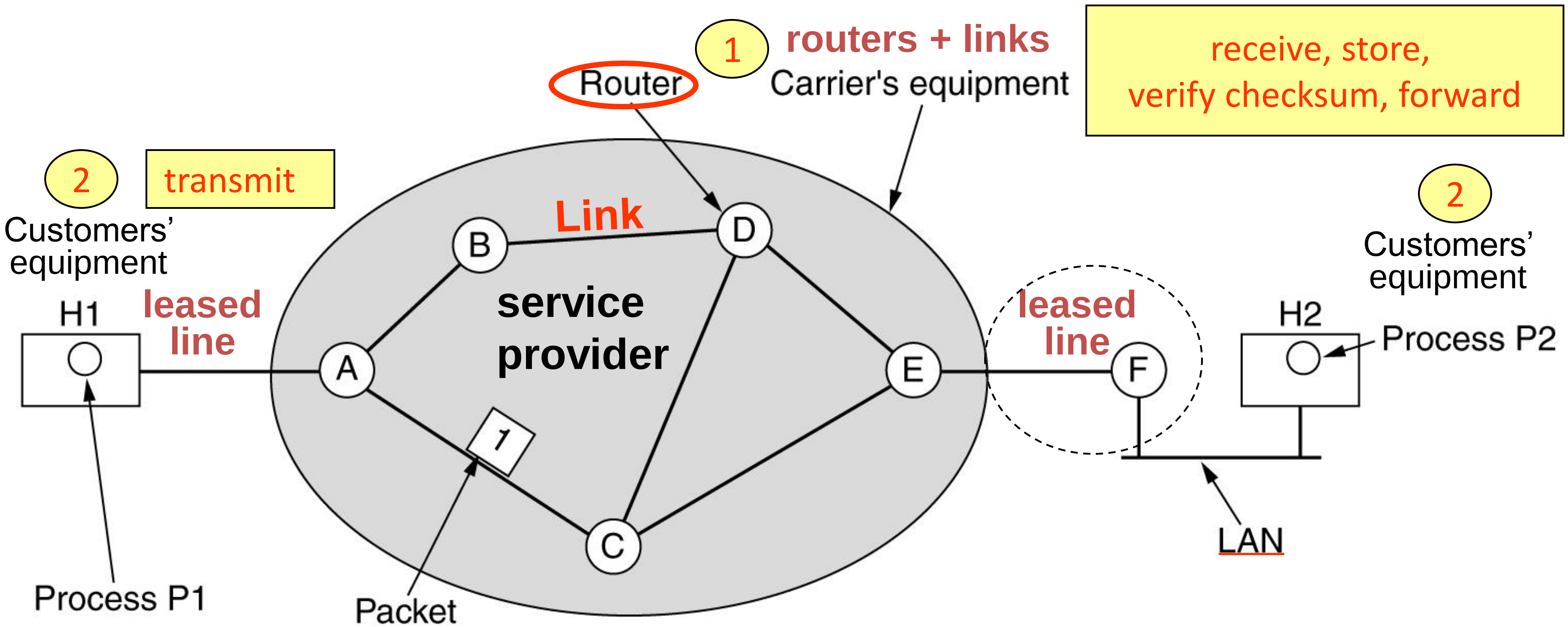
- **services provided to the transport layer**
- **internal design of the subnet**

- **store-and-forward packet switching**
- **services provided to the transport layer**
- **connection-less service**
- **connection-oriented service**
- **comparison of datagram and virtual-circuit subnets**

23.03.2020 17:11 4

Computer Networks
Network Layer

Design issues
Store-and-Forward Packet switching



Environment of network layer protocols

23.03.2020 17:11 5

Computer Networks

Network Layer

Design issues

Services provided to the transport layer

NL provides services to the TL at the NL-TL interface

NL services are designed to cater to :

- **services should be independent of router technology**
- **TL should be shielded from the number / type / topology of routers in the subnet**

23.03.2020 17:11 6

Computer Networks
<u>Network Layer</u>
Design issues

Services provided to the transport layer

NL services are designed to cater to :

- **network addresses made available to TL should use a uniform numbering plan, even across LANs and WANs**

connection-less service vs. connection-oriented service

23.03.2020 17:11 7

Computer Networks
Network Layer

Connectionless Service **(Ex. the Internet)**

packets moved to the subnet individually and routed independent of each other

no advance set up is needed

each packet must carry the full destination address

in this context, packets are called *datagrams* and the subnet is called *datagram subnet*

23.03.2020 17:11 8

Computer Networks
Network Layer

Connection-oriented Service **(Ex. ATM)**
path from source to destination is established before sending data packets - called Virtual Circuit or *VC*

subnet is called *VC subnet*

23.03.2020 17:11 9

Computer Networks

Network Layer

Connection - oriented Service

A virtual circuit :

- **is a circuit or path between points in a network**
- **appears to be a discrete, physical path, but …..**
- **is actually a managed pool of circuit resources from which …..**
- **specific circuits are allocated, as needed, to meet traffic requirements**

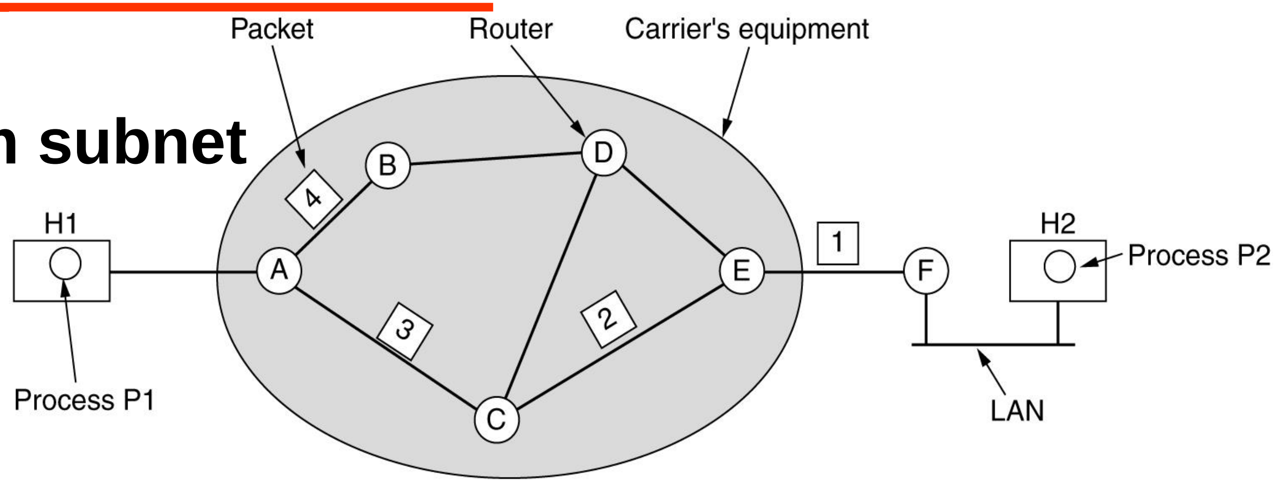
23.03.2020 17:11 10

Computer Networks

Network Layer

Connectionless Service

Datagram subnet



A's table initially (Dest.)	(Line)	A's table later (Dest.)	(Line)	C's table (Dest.)	(Line)	E's table (Dest.)	(Line)
A	–	A	–	A	A	A	C
B	B	B	B	B	A	B	D
C	C	C	C	C	–	C	C
D	B	D	B	D	D	D	D
E	C	E	B	E	E	E	–
F	C	F	B	F	E	F	F

Dest. Line

Router tables

- destination
- outgoing line

assume that message is 4 times longer than maximum packet size

network layer has to break it into 4 packets 1, 2, 3, 4

23.03.2020 17:11 11

Computer Networks

Network Layer

process P1 in host H1 is required to send a long message to P2 in H2

every router has an internal table with

table entries : (a) destination (b) outgoing line for that destination

packets 1, 2, 3 from H1 arrive at A, are stored (for error checking) at A before forwarding

packets from A arrive at F thro C and E, are *encapsulated* in a DLL frame and sent to H2 on LAN

23.03.2020 17:11 12

Computer Networks

Network Layer

packet 4 arriving at A may need to be treated differently, and sent to F through, say, path BDE

for this to happen, A takes a decision based on some analysis, updates its tables and sends the packet

***Routing algorithm* is required to**

(a) *manage* tables

(b) take *decisions*

23.03.2020 17:11 13

Computer Networks

Network Layer

Connection - oriented Service

virtual circuit (VC) subnet is needed

choosing a new route for every packet sent is avoided

when a connection is established, …

a route from source host to destination host is chosen

this is done as part of connection setup

connection details are stored in tables inside routers

23.03.2020 17:11 14

Computer Networks

Network Layer

Connection - oriented Service

the stored route is used for all data traffic flowing over the connection

analogous to telephone system

when connection is released, VC is terminated

each packet carries an id indicating which VC it belongs to

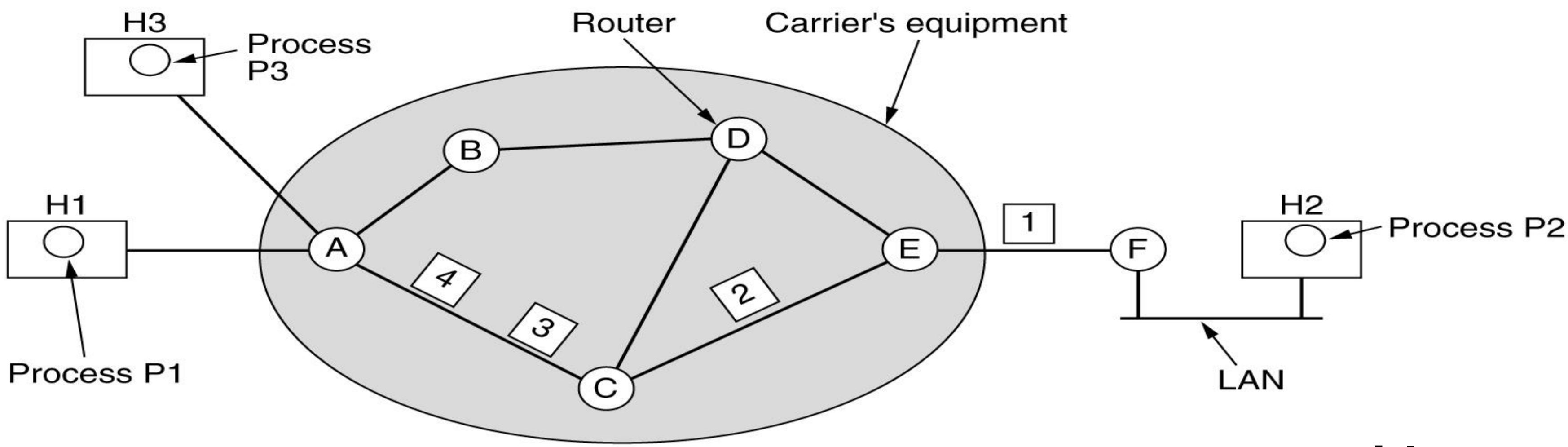
23.03.2020 17:11 15

Computer Networks

Network Layer

Connection - oriented Service

Virtual - Circuit (VC) subnet is required



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

Router tables

Router tables
- source / destination
- connection ID

23.03.2020 17:11

16

Computer Networks

Network Layer

Connection - oriented Service

A's table

In		Out	
H1	1	C	1
H3	1	C	$\boxed{2}$

C's table

A	1	E	1
A	2	E	2

E's table

C	1	F	1
C	2	F	2

host H1 has established connection-1 with host H2

1st line of A's table : if a packet bearing connection id 1 comes in from H1,

send it to C and give connection id 1

23.03.2020 17:11 17

Computer Networks

Network Layer

Connection - oriented Service

1st line of C's table : route packet to E, also with connection id 1

host H3 also establishes connection with H2, sends packet with *connection id 1*

2nd line of A's table : if a packet bearing connection id 1 comes in from H3, ….

send it to C and give *connection id 2*

(also called *label switching*)

23.03.2020 17:11 18

Computer Networks

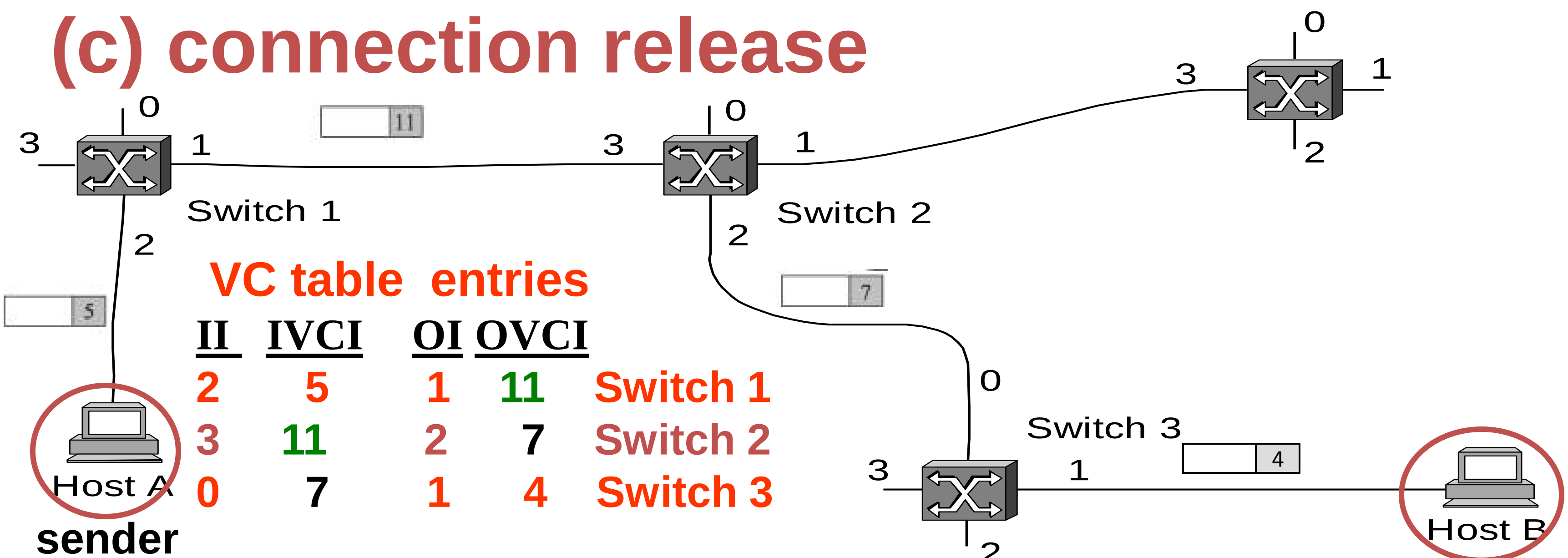
Network Layer

Connection - oriented Service

(a) setting up of VC between hosts before data is sent - this phase is called ***connection setup***

(b) packet transfer

(c) connection release



VC table entries

II	IVCI	OI	OVCI	
2	5	1	11	Switch 1
3	11	2	7	Switch 2
0	7	1	4	Switch 3

23.03.2020 17:11

19

Computer Networks

Connection setup phase :

establishment of *connection state* in each of the switches between source and destination host

connection state for a single connection consists of an entry in the VC table

VC table contains entries for each switch:

(a) virtual circuit identifier (VCI) :

uniquely identifies the connection, carried inside header of packets belonging to this connection

23.03.2020 17:11 20

Computer Networks

VC table contains entries for each switch:

(b) incoming interface (II) on which packets arrive at the switch

(c) outgoing interface (OI) on which packets leave the switch

(d) a different VCI (OVCI) is used for outgoing packets

23.03.2020 17:11 21

Computer Networks

Network Layer

VC :

choose route from source to destination

set up connection

add identifier (*VC number*) to each packet (router replaces connection ids in outgoing packets - called *label switching*)

store details in router tables

use the route for transporting all packets

release connection, terminate VC

23.03.2020 17:11 22

Computer Networks
Network Layer

Comparison of Datagram and VC subnets

Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed **complicated lookup**	Required **setup time**
Addressing	Each packet contains the full source and destination address **bandwidth**	Each packet contains a short VC number **router memory**
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently **easy traffic load balancing**	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets in queue 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 **dynamic allotment of CPU cycles, buffers, bandwidth**	Easy if enough resources can be allocated in advance for each VC

23.03.2020 17:11 23

Computer Networks
Network Layer

Example **(Cost of VC vs datagram)** AT 5.5

Consider the following two cases :

(i) a VC service implementation using 3-byte header and 8 bytes of storage for VCid

(ii) a datagram implementation service using 15-byte header

The average duration of a session is 1000 seconds during which 200 packets are transmitted. The average no. of hops is 4.

23.03.2020 17:11 24

Example

AT 5.5

Assume :

(i) cost of transmission capacity is 0.5 paisa per 10^6 bytes per hop

(ii) cost of router memory is 0.5 paisa per byte and depreciates in 2 years @ 40 hours per week

Compute and compare the cost of *overheads* in the above two implementations (VC and datagram) for the given set of parameters

23.03.2020 17:11 25

Computer Networks

Network Layer

Example **AT 5.5**

Computation of cost of Memory :

Duration of working in 2 years

$= (2 \times 52 \times 40 \times 3600) = 1.5 \times 10^7$ seconds

Cost per byte-second

$= (0.5 \text{ paisa} \div 1.5 \times 10^7) = 0.33 \times 10^{-7}$ paisa

23.03.2020 17:11 26

Computer Networks

Network Layer

Example AT 5.5

No. of routers required for 4 hops = 5

VC implementation :

Router memory required = 5 x 8 = 40 bytes

Session duration = 1000 seconds

Memory requirement = 40000 byte-secs

Cost = $4 \times 10^{4} \times 0.33 \times 10^{-7}$ paisa

= 1.33×10^{-3} paisa

23.03.2020 17:11 27

Computer Networks
Network Layer

Datagram implementation :

No. of bytes of additional header (over the VC case implementation) to be transmitted

= 12 per packet per hop = 12 x 200 x 4

= 9600 byte-hops

Transmission cost = 0.5×10^{-6} per byte-hop,

Cost = $9600 \times 0.5 \times 10^{-6}$ paisa

= 4.80×10^{-3} paisa

Compared with 1.33×10^{-3} paisa for the VC case, Datagram implementation is costlier for the given set of parameters

23.03.2020 17:11 28

Computer Networks

Network Layer

Routing algorithms

- Shortest Path Routing
- Flooding
- Distance Vector Routing
- Link State Routing
- Hierarchical Routing
- Broadcast Routing
- Multicast Routing

23.03.2020 17:11 29

Computer Networks

Network Layer

Routing algorithms

routing algorithm **is a part of the Network Layer software that is responsible for**

deciding **……….**

which output line should be used to transmit or *forward* an incoming packet at a router

23.03.2020 17:11 30

Computer Networks

Network Layer

Routing algorithms

Case of datagram : decision to be made *afresh* every time a packet arrives, since the *best* route can change dynamically

Case of VC : routing decision to be made when a new VC is to be set up
route remains in force for an entire *session* (say *ftp* or *remote login* or *telnet*)
-- called *session routing*

23.03.2020 17:11 31

Computer Networks
Network Layer

Two processes inside Router

Routing	Forwarding
filling in and updating .... routing tables use of *routing algorithm*	handling packet as it arrives, i.e. looking up routing table to ascertain out - going line and send packet

23.03.2020 17:11 32

Computer Networks

Network Layer

Desirable features in routing algorithm

- correctness
- simplicity
- robustness (tolerance(#) to faults, topology & traffic changes, router crashes)
- stability (quick convergence to equilibrium)
- fairness
- optimality

(#) : without requiring all jobs in all hosts to be aborted and the network to be rebooted every time some router crashes

23.03.2020 17:11
33

Computer Networks

Network Layer

classes of routing algorithm :

- **non-adaptive or static**
- **adaptive or dynamic**

23.03.2020 17:11
34

Computer Networks

Network Layer

Non-adaptive algorithm : (*static routing*)

routing decisions are not based on measurements or estimates of current traffic or topology

choice of route is computed in advance, off-line and ….

downloaded to routers at boot time

23.03.2020 17:11 35

Computer Networks

Network Layer

Adaptive algorithm : (*dynamic routing*)

routing decisions are changed commensurate with topology and traffic changes

adaptive algorithms differ in :

- **where they get info : from neighbouring or all routers**
- **when they change routes : every few msec or when load or topology changes**
- **optimization metrics used : distance or number of hops or estimated transit time**

23.03.2020 17:11 36

Computer Networks

Network Layer

Flooding (*static method*)

every incoming packet is sent on every *outgoing* line, except the one it arrived on

generates large numbers of duplicate packets endlessly

so, measures are required to damp the process

23.03.2020 17:11
37

Computer Networks

Network Layer

Flooding (*static method*)

damping the process of flooding : (a)
a hop counter (ideally, initialized to source-to-destination path length) is kept in the header of each packet

count is decremented each hop

packet is discarded when hop count = 0

23.03.2020 17:11 38

Computer Networks

Network Layer

Flooding (*static method*)

damping the process of flooding : (b)

source router introduces a sequence number in each packet it receives from its hosts

each router maintains a list, per source router, containing which sequence numbers originating at that source has been 'seen'

if incoming packet serial number is already on the list , it is not flooded

23.03.2020 17:11 39

Computer Networks

Network Layer

Flooding (*static method*)

damping the process of flooding : (c)

Selective flooding

routers do not send every incoming packet out every time

send only on those lines that are going approximately in the right direction

23.03.2020 17:11 40

Computer Networks

<u>Network Layer</u>

<u>Flooding</u> (*static method*) : Uses

Flooding is not practical in most applications, but is highly robust and so, is desirable in :

- ***military* : disaster recovery / fault tolerance requirements**

- ***distributed database :* especially during updation of all databases concurrently**

23.03.2020 17:11

41

Computer Networks

Network Layer

Distance Vector Routing (dynamic method)

Bellman - Ford / Ford - Fulkerson Algorithm

each router maintains a table (i.e. vector), indexed by, and containing one entry for, each router in the subnet; each entry has two parts :

(a) the best known distance to each destination

{ metric could be number of hops, time delay,(#) total no. of packets in queue along the path, … }

(b) preferred outgoing line to use to reach there

{ table is updated by exchanging information with *neighbours* }

(#) : by use of ECHO packets

23.03.2020 17:11 42

Network Layer

Distance Vector Routing

each router takes the following actions :

(a) it counts the *weight* of the links directly connected to it and saves the information to its table

(b) in a specific period of time, it send its table to its *neighbour* routers (not to all routers) and receives the routing table of each of its neighbours

(c) based on the information in its neighbours' routing tables, it updates its own

23.03.2020 17:11 43

Computer Networks

Network Layer

Distance Vector Routing

Metrics used : no. of hops (neighbour = 1)

queue length (exam each queue)

delay (use ECHO packets)

assume that delay is used as metric and router knows delay to each of its neighbors:

every T msec, each router sends to each neighbour a list of its estimated delays to each destination and also receives a similar list from each neighbour

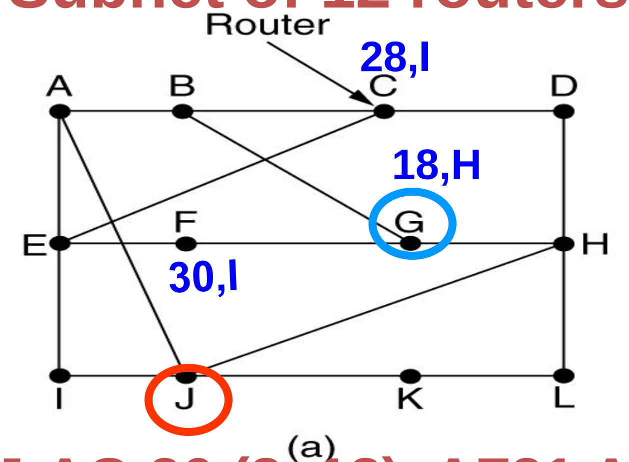
23.03.2020 17:11 44

Computer Networks

Network Layer

Distance Vector Routing

Subnet of 12 routers



delay vectors received from the neighbours of J

New estimated delay from J

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)

measured or estimated delay by J

J-AG 26 (8+18) AF31 AC33

J-IG 41 (10+31) IF30 IC28

J-HG 18 (12+6) HF31 HC31

J-KG 37 (6+31) KF46 KC42

Same calculation is performed for all other destinations

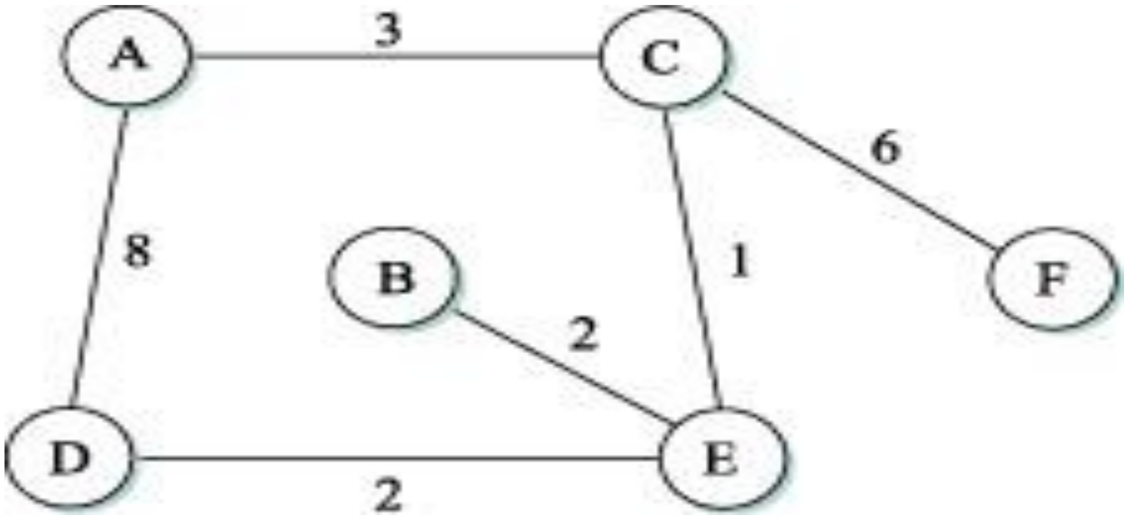
23.03.2020 17:11

45

Computer Networks

Example : Network Layer

PD 3.3



For the network shown above, give the datagram forwarding table (destination, next hop) for each node. The links are labelled with relative costs.

Criteria : each packet should be forwarded via the lowest-cost path to its destination.

23.03.2020 17:11 46

Example : Network Layer

PD 3.3

A

Destination	Next Hop
B	C
C	C
D	C
E	C
F	C

B

Destination	Next Hop
A	E
C	E
D	E
E	E
F	E

C

Destination	Next Hop
A	A
B	E
D	E
E	E
F	F

D

Destination	Next Hop
A	E
B	E
C	E
E	E
F	E

Forwarding Tables

23.03.2020 17:11

47

Computer Networks

Example : Network Layer

PD 3.3

E

Destination	Next Hop
A	C
B	B
C	C
D	D
F	C

F

Destination	Next Hop
A	C
B	C
C	C
D	C
E	C

Forwarding Tables

23.03.2020 17:11 48

Computer Networks

Example : Network Layer

PD 3.3

Node A:

Destination	Next hop
B	C
C	C
D	C
E	C
F	C

Node B:

Destination	Next hop
A	E
C	E
D	E
E	E
F	E

Node C:

Destination	Next hop
A	A
B	E
D	E
E	E
F	F

Node D:

Destination	Next hop
A	E
B	E
C	E
E	E
F	E

Node E:

Destination	Next hop
A	C
B	B
C	C
D	D
F	C

Node F:

Destination	Next hop
A	C
B	C
C	C
D	C
E	C

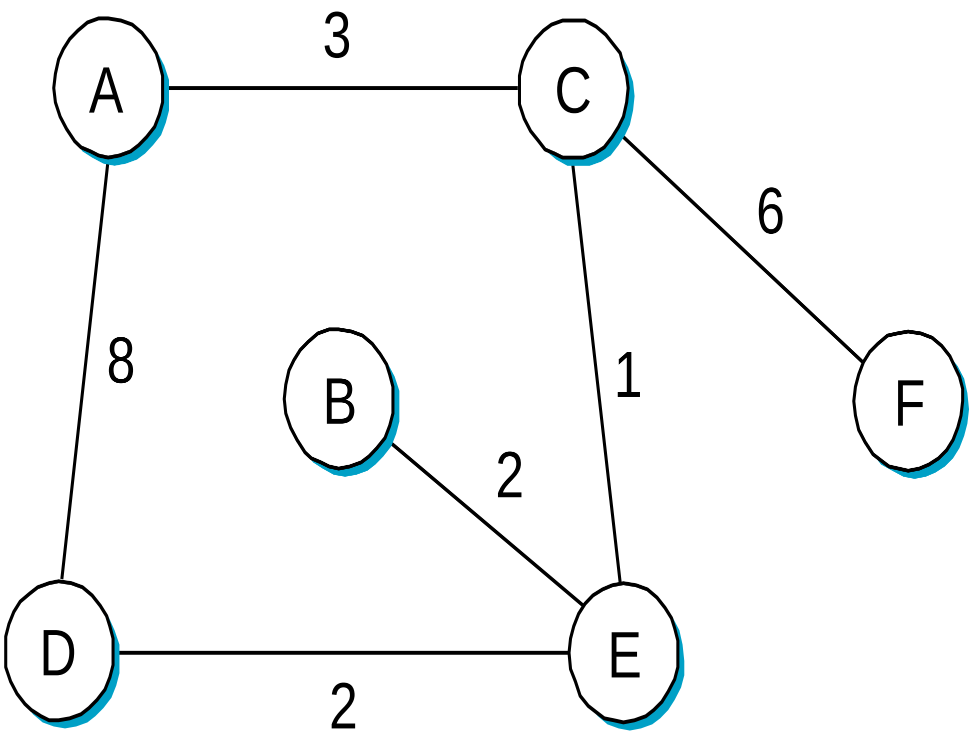
23.03.2020 17:11

Computer Networks

Distance Vector Routing :

Example (distance)

PD 4.15



Work out DV tables when :

(a) each node knows the distances to its immediate neighbours

(b) each node has reported the info it had in (a) to its immediate neighbours

(c) step (b) happens a second time

23.03.2020 17:11 50

Computer Networks

Distance Vector Routing :

Example (distance)

PD 4.15

Information Stored at Node	Distance to Reach Node					
	A	B	C	D	E	F
A						
B						
C						
D						
E						
F						

Work out DV tables when :

(a) each node knows the distances to its immediate neighbours

(b) each node has reported the info it had in (a) to its immediate neighbours

(c) step (b) happens a second time

23.03.2020 17:11
51

Computer Networks

Distance Vector Routing :

Example

PD 4.15

(a)

Information Stored at Node	Distance to Reach Node					
	A	B	C	D	E	F
A	0	∞	3	8	∞	∞
B	∞	0	∞	∞	2	∞
C	3	∞	0	∞	1	6
D	8	∞	∞	0	2	∞
E	∞	2	1	2	0	∞
F	∞	∞	6	∞	∞	0

23.03.2020 17:11
52

Computer Networks

Distance Vector Routing :

Example

PD 4.15

(b)

Information Stored at Node	Distance to Reach Node					
	A	B	C	D	E	F
A	0	∞	3	8	4	9
B	∞	0	3	4	2	∞
C	3	3	0	3	1	6
D	8	4	3	0	2	∞
E	4	2	1	2	0	7
F	9	∞	6	∞	7	0

23.03.2020 17:11 53

Computer Networks

Distance Vector Routing :

Example

PD 4.15

(c)

Information Stored at Node	Distance to Reach Node					
	A	B	C	D	E	F
A	0	6	3	6	4	9
B	6	0	3	4	2	9
C	3	3	0	3	1	6
D	6	4	3	0	2	9
E	4	2	1	2	0	7
F	9	9	6	9	7	0

23.03.2020 17:11 54

Computer Networks

Network Layer

Distance Vector Routing   Example   AT 5.9

Consider the subnet shown in the figure (next slide).

Distance vector routing is used, and the following vectors have come in to router C :

from B : (5, 0, 8, 12, 6, 2)

from D : (16, 12, 6, 0, 9, 10)

from E : (7, 6, 3, 9, 0, 4)

The measured delays to B, D and E are 6, 3 and 5 respectively.

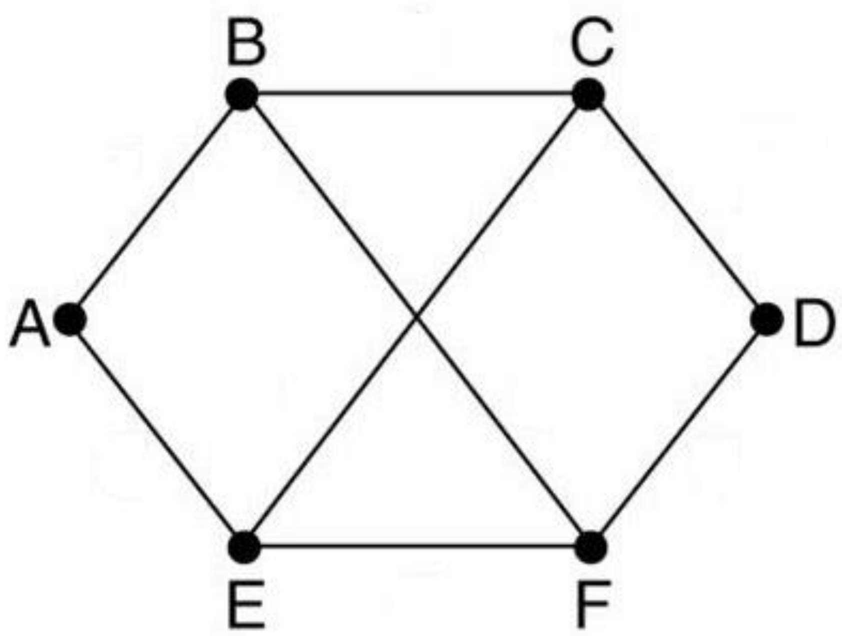
Give C’s new routing table indicating outgoing line to use and the expected delay.

23.03.2020 17:11 55

Computer Networks

Network Layer

Distance Vector Routing Example AT 5.9



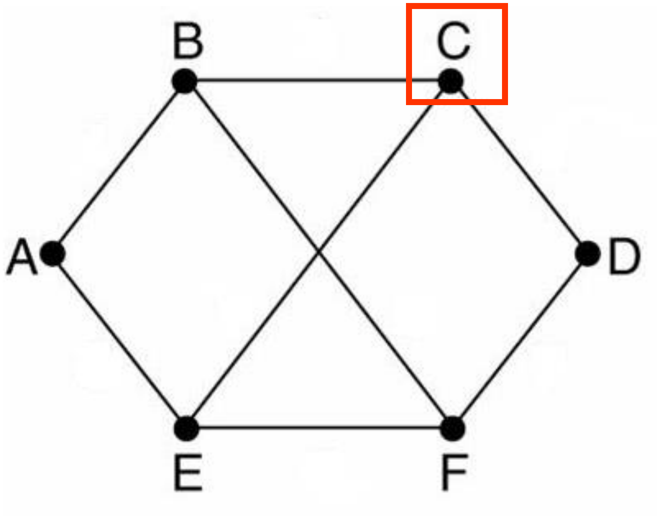
23.03.2020 17:11 56

Computer Networks

Network Layer

Distance Vector Routing : Example

AT 5.9



Vectors coming into C from →	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
Measured Delay C to B/D/E	6	3	5

23.03.2020 17:11
57

Computer Networks

Network Layer

Example AT 5.9

{r}	B	D	E	C - B - {r}	C - D - {r}	C - E - {r}	Line
A	5	16	7				
B	0	12	6				
C	8	6	3				
D	12	0	9				
E	6	9	0				
F	2	10	4				
Measured Delay C-B/D/E	6	3	5				

23.03.2020 17:11
58

Computer Networks

Network Layer

Example AT 5.9

{r}	B	D	E	C - B - {r}	C - D - {r}	C - E - {r}	Line
A	5	16	7	6+5=11	3+16=19	5+7=12	B
B	0	12	6	6+0=6	3+12=15	5+6=11	B
C	8	6	3	6+8=14 (0)	3+6=9 (0)	5+3=8(0)	-
D	12	0	9	6 +12=18	3	5+9=14	D
E	6	9	0	6+6=12	3+9=12	5	E
F	2	10	4	6+2=8	3+10=13	5+4=9	B
Measured Delay C-B/D/E	6	3	5				

23.03.2020 17:11
59

Computer Networks

Network Layer

Distance Vector Routing : converges, but *slowly*

The count-to-infinity problem

metric used : number of hops

A	B		C	D	E	
	•	∞	•	•	•	Initially
	1		•	•	•	After 1 exchange
	1		2	•	•	After 2 exchanges
	1		2	3	•	After 3 exchanges
	1		2	3	4	After 4 exchanges

(a)

- A is initially down ;
- entries by other nodes is infinity;
- A comes up

	X A	B	C	D	E	
		1	2	3	4	Initially
B		3	2	3	4	After 1 exchange
C		3	4	3	4	After 2 exchanges
D		5	4	5	4	After 3 exchanges
E		5	6	5	6	After 4 exchanges
		7	6	7	6	After 5 exchanges
		7	8	7	8	After 6 exchanges
			⋮			
		•	•	•	•	

(b)

A is initially up ; later goes down
(or link A-B cut)

23.03.2020 17:11
60

Computer Networks

Network Layer

Distance Vector Routing : demerits

(a) the delay metric used in ARPANET was queue length;

line bandwidths were not taken into account;

when low bandwidth lines (56 Kbps) were upgraded to higher values (230 Kbps, 1.544 Mbps) …. caused problems

(b) count-to-infinity problem

23.03.2020 17:11 61

Computer Networks

Network Layer

Complete topology and all delays are experimentally measured and distributed to *every* router in the subnet

Link State Routing

Each router :

(a) discovers its neighbours and learns their network addresses

(b) measures delay (or cost) to each of its neighbours (have a reasonable estimate)

(c) constructs a packet containing addresses and delays (or costs)

(d) sends this packet to all other routers

(e) computes the shortest path to every other router (Dijkstra's algorithm)

23.03.2020 17:11 62

Computer Networks

Network Layer

Link State Routing : (a) Learning about neighbours

First task after booting - learn who the neighbours are

Sends a special *HELLO* packet on each point-to-point line

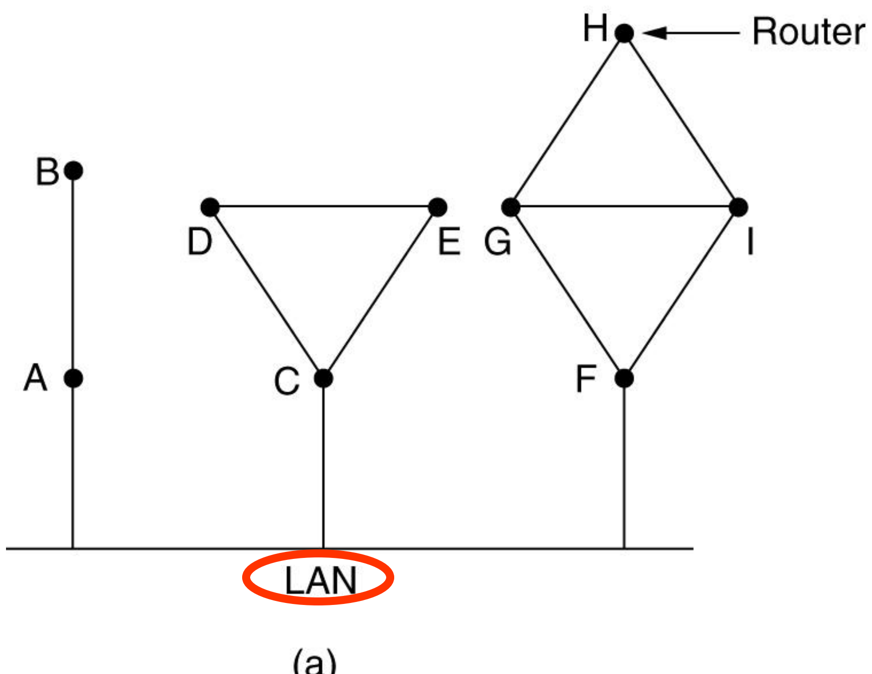
Expects response (from router at other end) giving its *identity*

23.03.2020 17:11 63

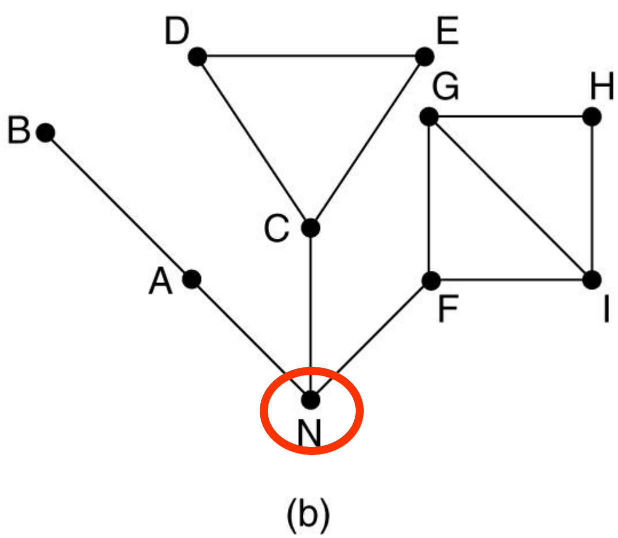
Computer Networks

Network Layer

Link State Routing : (a) Learning about neighbours



routers A, C , F connected to a LAN



graph model of (a) with LAN modelled as node N

23.03.2020 17:11
64

Computer Networks

Network Layer

Link State Routing : (b) Measuring line cost

Send *ECHO* packet

Other router is required to respond

Measure round-trip delay ; divide by 2

For better results, repeat test, use average

Issue : take load into account ?

Yes - start round-trip timer when *ECHO* packet is queued (treats lightly loaded path as shorter)

No - start round-trip timer when *ECHO* packet reaches the front of queue

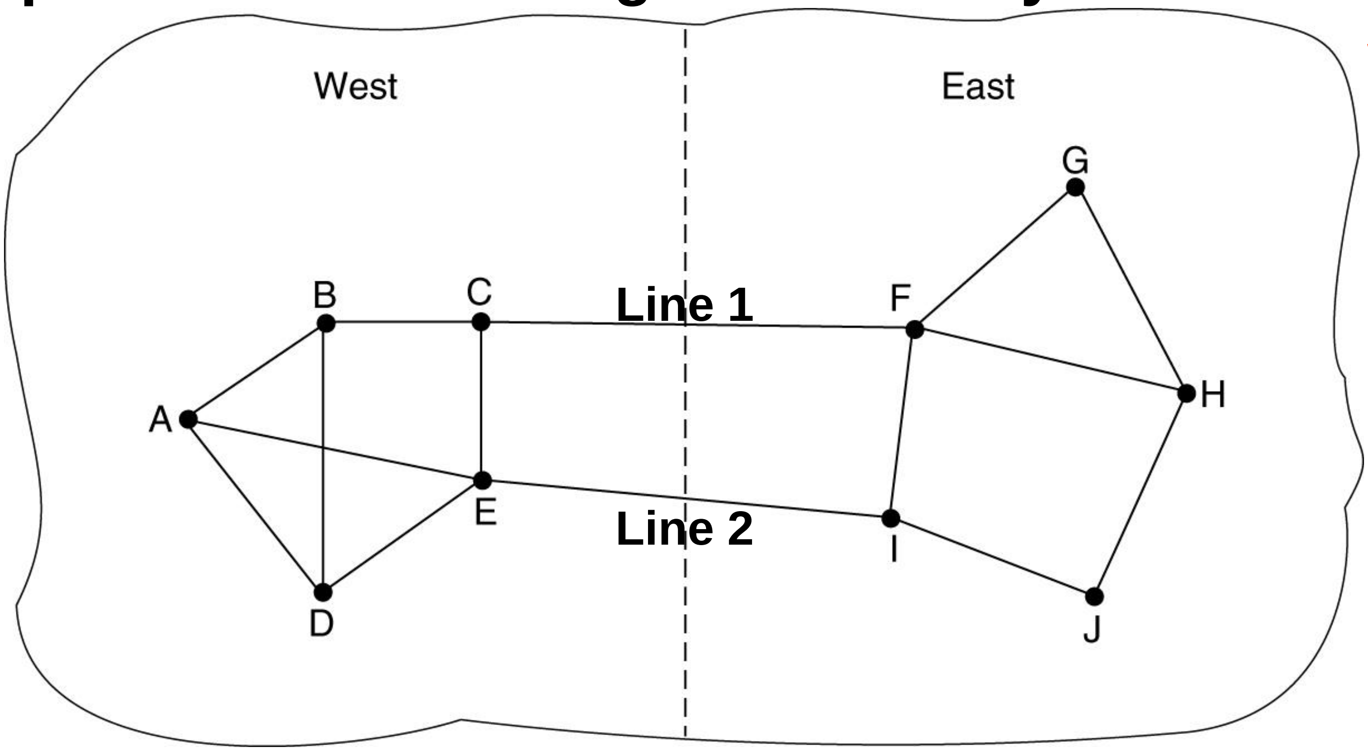
23.03.2020 17:11
65

Computer Networks

Network Layer

Link State Routing : (b) Measuring line cost

problem of including load in delay calculations



wild oscillations of routing tables, erratic routing

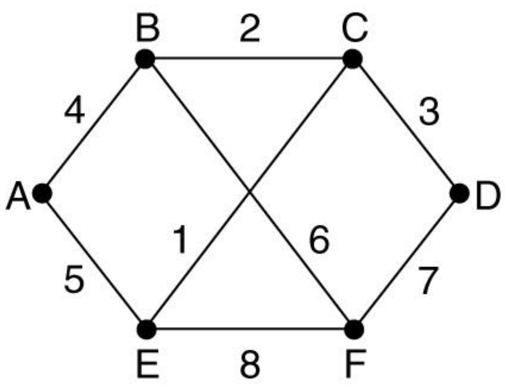
solution : distribution of load over multiple lines

23.03.2020 17:11
66

Computer Networks

Network Layer

Link State Routing : (c) Constructing Link State Packets



(a)

Link State Packets

	A		B		C		D		E		F	
identity	A		B		C		D		E		F	
seq. no.	Seq.		Seq.		Seq.		Seq.		Seq.		Seq.	
age	Age		Age		Age		Age		Age		Age	
list of neighbors	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
	L	D										

(#)

(b)

Information collected regarding *neighbours* is to be sent to *all other* routers

Action required : construction of packets; when to construct ?

(#) → L : List of neighbours ; D : Delay

67

Link State Routing : (c) Constructing Link State Packets

Packets can be built : (when ?)

at regular intervals

event-driven
(ex. neighbour / link going down / coming up / change of properties)

23.03.2020 17:11 68

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets

Need : *reliable* distribution ---- issue as the packets are distributed and installed, routers getting the first LS packets will change their routes → over a period of time, different routers may use different versions of topology → inconsistencies, loops, unreachable machines, ……

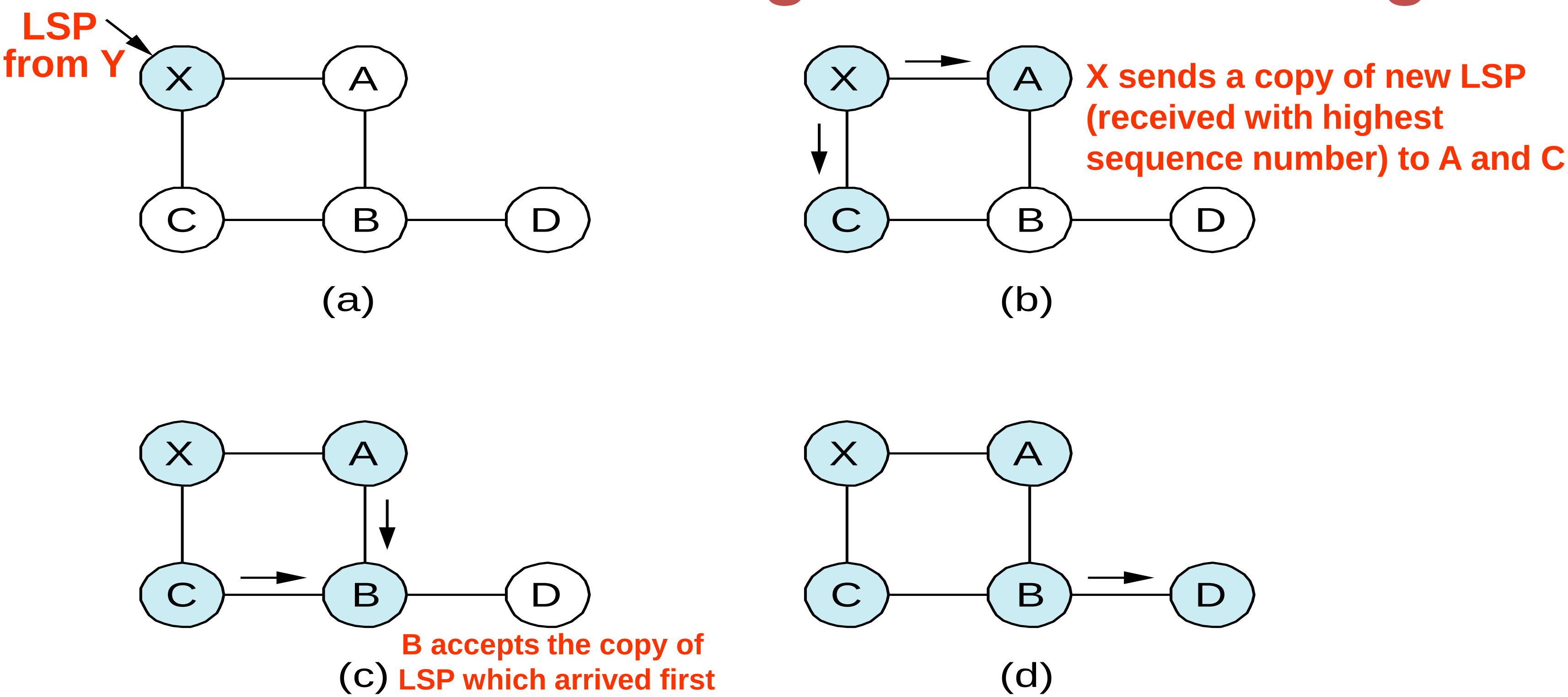
23.03.2020 17:11 69

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets

Basic distribution algorithm : flooding



23.03.2020 17:11 70

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets

Basic distribution algorithm : flooding

Why Seq. no. ? → to keep control of the flood :
each packet contains a sequence no.
sequence no. is incremented for each new packet sent
other routers keep track of the following information :
{source router & sequence no.} →

23.03.2020 17:11
71

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets

When a *new* LS packet comes in : check against list of all packets already 'seen' →

(a) new ? → forward to all lines except on which it arrived

(b) duplicate ? → discard

(c) sequence no. is lower than highest no. already seen ? → treat it as obsolete and discard

Issues? → →

23.03.2020 17:11 72

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets

Issues :

(a) Sequence numbers *wrapping around* → confusion

Solution : use a 32-bit sequence number ;

@ 1 LS packet per sec, wrap around time = 137 years ; this possibility can be ignored

(b) Router crash : sequence is broken, set to 0 at reboot, next packet rejected as duplicate

23.03.2020 17:11 73

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets

Issues :

(c) If sequence no. is corrupted i.e. received with (even 1 bit) error, packets will be rejected as obsolete

e.g. number 4 is received as 65,540 (#) ? → packets 5 through 65,540 will be rejected as obsolete, since current sequence number is thought to be 65,540

(#) 65540 = 00000000000000010000000000000100

23.03.2020 17:11

74

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets

Issues :

(c) If sequence no. is corrupted….

Solution :

include *age* of packet after the sequence no.

decrement it once per second

when age becomes zero, discard the link state packet from that router

23.03.2020 17:11 75

Computer Networks

Network Layer

Link State Routing : (e) Computation of new route

(i) once a router has accumulated a full set of LSPs …..
it constructs the entire subnet graph, because every link is represented

(ii) Dijkstra's algorithm is run locally to construct the shortest path to all possible destinations;

23.03.2020 17:11 76

Computer Networks

Network Layer

Link State Routing : (e) Computation of new route

(iii) results of this algorithm is installed in the routing tables

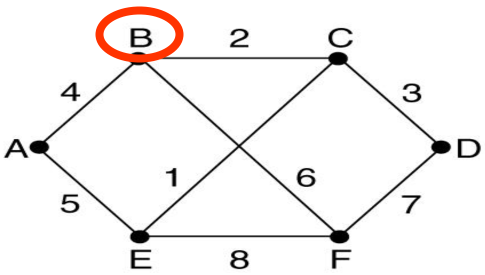
(iv) normal operation is resumed

23.03.2020 17:11 77

Computer Networks

Network Layer

Link State Routing : (d) Distributing Link State Packets



(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)

Packet buffer for Router B

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	

23.03.2020 17:11 78

Computer Networks

Network Layer

Hierarchical Routing

growth of Networks in size →

increase in routing tables →

increase in RAM (to hold tables) CPU time (to scan the tables) / bandwidth (to send status reports)

→ infeasibility of every router having info about every other router →

routing is better accomplished if done hierarchically e.g. as in telephone network

23.03.2020 17:11 79

Computer Networks

Network Layer

Hierarchical Routing

routers are divided into *regions*

each router is concerned about routers only within *its* region

need not be concerned about the internal structure of *other* regions

for large networks, an n-level hierarchy may be necessary

23.03.2020 17:11 80

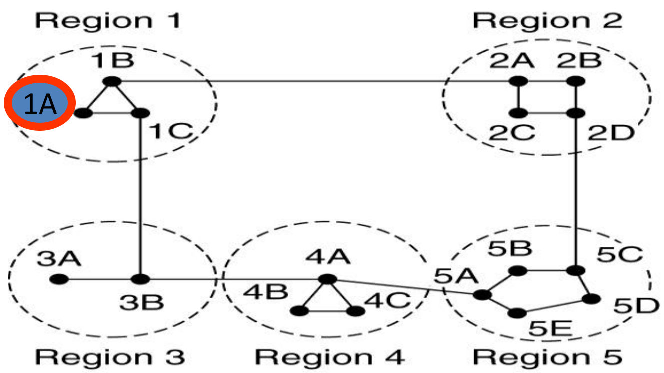
Computer Networks

Network Layer

Hierarchical Routing

routers.. regions.. clusters.. zones.. groups

two-level hierarchy with five regions :



↑regions ↑path lengths
(1A to 5C)

(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

17 entries

(b)

Hierarchical table for 1A

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

7 entries

(c)

23.03.2020 17:11
81

Computer Networks

Network Layer

Example : 720 routers **(how many levels of hierarchy ?)**

No hierarchy → each router needs 720 routing table entries

- **24 regions x 30 routers → 30 local + 23 = 53 remote entries**
- **8 clusters x 9 regions x 10 routers → 7 for clusters + 8 for regions + 10 local = 25**

Optimal no. of levels for an *N* router subnet = *ln N* **(due to Kamoun and Kleinrock,1979)**

23.03.2020 17:11 82

Computer Networks
Network Layer

Example : AT 5.12

For hierarchical routing with 4800 routers, what region & cluster sizes should be chosen to minimize the size of routing table for a 3-layer hierarchy ?

Assume that a solution with *k* clusters of *k* regions of *k* routers is close to optimal i.e. $k \approx (4800)^{1/3}$

Use trial and error to check out the combination , where all the three parameters (clusters + regions + routers) are in the general vicinity of *k*

23.03.2020 17:11 83

Computer Networks

Network Layer

Example : AT 5.12

One combination :

15 clusters, with each cluster having …….

16 regions , with each region having …….

20 routers

Total = 51 entries

23.03.2020 17:11 84

Computer Networks
Network Layer

Broadcast Routing :

hosts may need to send messages to many destinations simultaneously so that those interested read the data (#)

Methods :

- **sending a distinct packet to each destination**
- **flooding**
- **multi-destination routing**
- **spanning tree routing**
- **reverse path forwarding**

(#) **live programs, weather reports, data updates**

23.03.2020 17:11 85

Broadcast Routing :

sending a packet to all destinations simultaneously … called *broadcasting*

1. **Send a distinct packet to *each destination* :**
   - **→ waste of bandwidth and …..**
   - **→ source needs to have a full list of all destinations**
2. ***Flooding* → generation of too many packets, consumption of too much bandwidth**

23.03.2020 17:11 86

Computer Networks

Network Layer

Broadcast Routing :

3. Multi-destination routing

- **each packet contains a list of desired destinations**
- **when a packet arrives at a router, router checks all destinations**
- **router determines the set of output lines needed (best route to at least one destination)**

23.03.2020 17:11 87

Computer Networks

Network Layer

Broadcast Routing :

3. Multi-destination routing

- **router generates a new copy of the packet for each output line to be used and includes, in each packet, only those destinations that are to use the line**
  i.e. the destination set is partitioned among the output lines

23.03.2020 17:11 88

Computer Networks

Network Layer

Broadcast Routing :

4. Spanning-tree routing

a *spanning tree* is a subset of the subnet ; includes all routers, but contains *no loops*

each router copies an incoming broadcast packet onto all the spanning tree (ST) lines, except the incoming line

this method generates absolute minimum no. of packets required and makes excellent use of bandwidth

Issue : Router must have knowledge of ST

23.03.2020 17:11 89

Broadcast Routing :

5. Reverse path forwarding

a router forwards only one copy of packet; the others are dropped

the copy that is forwarded is the one that has travelled the shortest path from the source to the router

23.03.2020 17:11 90

Computer Networks

Network Layer

Broadcast Routing :

5. Reverse path forwarding

a broadcast packet arrives at router → router checks to see if the packet arrived on the line normally used to *send* packets *to* the source of broadcast →

if yes → highly probable that the broadcast itself followed the best route → therefore, it is the first copy to arrive →

23.03.2020 17:11 91

Computer Networks

<u>Network Layer</u>

<u>Broadcast Routing</u> :

5. Reverse path forwarding

in this case, router forwards copy of this packet to all lines except *the* incoming line

if the broadcast packet arrived on any other line, the packet is discarded as likely duplicate

23.03.2020 17:11 92

Computer Networks
Network Layer

Broadcast Routing :

5. Reverse path forwarding

Sink tree : definition

the set of optimal routes from all sources to a given destination form a *sink tree* rooted at the destination

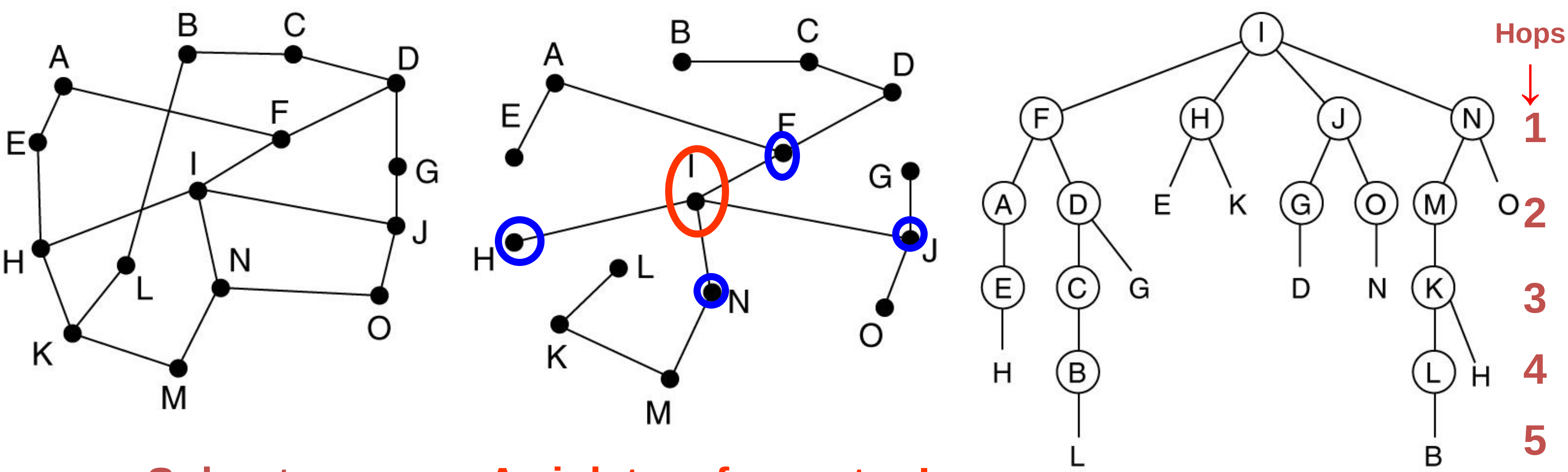
23.03.2020 17:11 93

Computer Networks

Network Layer

Broadcast Routing :

5. Reverse path forwarding



Subnet
(a)

A sink tree for router I
(b)

(c)
Tree built by reverse path forwarding

23.03.2020 17:11
94

Computer Networks

Network Layer

Broadcast Routing :

5. Reverse path forwarding

hop - 1 : I sends packets to F, H, J, N (four)

hop - 2 : eight packets are generated - five arrive on preferred lines (A, D, G, O, M)

hop - 3 : six packets are generated - three arrive on preferred lines (E, C, K)

hop - 4 : four packets are generated - two arrive on preferred lines (B, L)

total : 4 hops and 14 packets

(compare with 5 hops and 24 packets)

23.03.2020 17:11
95

Computer Networks
Network Layer

Broadcast Routing :

5. Reverse path forwarding

Advantages:

- **ease of implementation, efficient**
- **does not require routers to know about STs (as in ST routing)**
- **does not have overhead of a destination list (as in multi-destination routing)**
- **does not require any special process for damping (as in flooding)**

23.03.2020 17:11 96

Computer Networks

Network Layer

multicasting :
sending message
to a selected group

Multicast Routing :

a group of processes working together for implementing a common function

Example :
a distributed
database system

small groups may use point-to-point messaging

use of point-to-point messaging for large groups is expensive

multicast routing method involves sending messages to *well-defined groups,* large in size, but far less than the whole network

23.03.2020 17:11 97

Computer Networks

Network Layer

Multicast Routing :

Requires group management

methods are needed to :

- **create / destroy group**
- **to allow processes to join / leave groups**

when a process joins / changes group, it informs its host about this

23.03.2020 17:11 98

Multicast Routing :

routers should learn and know which host belongs to which group :

- **hosts should inform their routers about change of group**

or

- **routers should query hosts periodically**

23.03.2020 17:11 99

Computer Networks

Network Layer

Multicast Routing : pruning of spanning trees

each router computes a spanning tree covering all other routers

when a process sends a multicast packet to a group, …..

the first router examines its spanning tree

it *prunes* the tree by *removing* all lines that *do not lead* to hosts that are members of the group; use of *PRUNE* message (i.e. retains only required lines)

Multicast packets are forwarded only along the appropriate spanning tree

23.03.2020 17:11
100

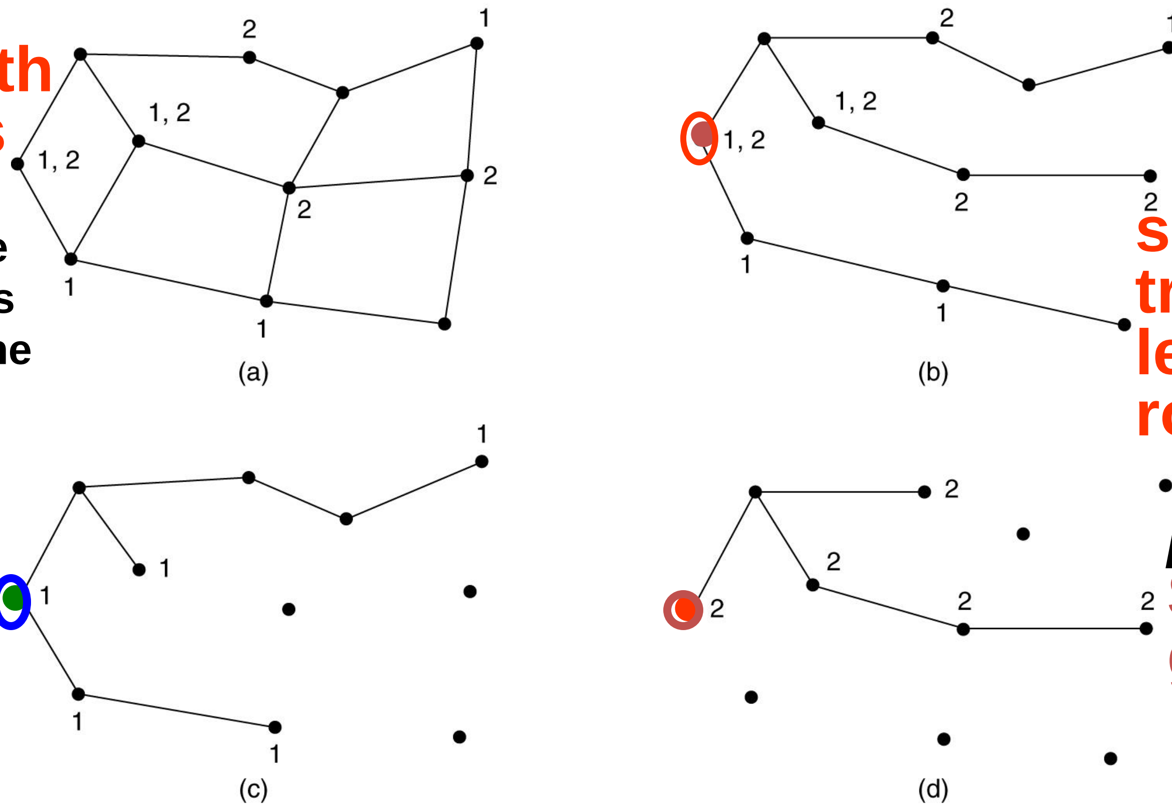
Computer Networks

Network Layer

Multicast Routing : each router computes a spanning tree covering all other routers

network with two groups 1 and 2

some routers are attached to hosts that belong to one or both groups



spanning tree for left-most router

pruned **ST for group 1**

pruned **ST for group 2**

23.03.2020 17:11
101

Computer Networks

Network Layer

Multicast Routing : pruning of spanning trees

Link state case :

each router is aware of complete topology, including host-group relationship

- **start at the end of each path**
- **work towards root**
- **remove all routers not belonging to the group in question**

23.03.2020 17:11 102

Computer Networks

Network Layer

Multicast Routing : pruning of spanning trees

Distance vector case :

basic algorithm : reverse path forwarding

when a router with :

(a) no hosts interested in a *particular group* and

(b) no connection to other routers

receives a multicast message for *that* group, it responds with a *PRUNE* message, informing sender not to send any more multicasts for *that* group

23.03.2020 17:11 103

Computer Networks

Network Layer

Multicast Routing : pruning of spanning trees

Distance vector case :

when a router with :

no group members among its own hosts has received multicast messages on all its lines

it will also respond with *PRUNE* message → the subnet is recursively pruned

23.03.2020 17:11 104

Computer Networks

Network Layer

Multicast Routing : disadvantage

poor scalability

assume :

a network has *n* groups

each group has an average of *m* members

for each group, *m* pruned STs must be stored; i.e a total of *mn* trees

when many groups exist, considerable storage is needed to store all the trees

23.03.2020 17:11 105

Computer Networks

Network Layer

Multicast Routing : alternative design

Core-based trees method

a single spanning tree per group is computed

the root (or the core) is placed near the middle of the group

to send multicast message :

- **host sends to the core**
- **core multicasts along spanning tree**

Storage required is for only one tree per group

23.03.2020 17:11
106