NETWORK LAYER

Syllabus

- Network layer design issues. Routing algorithms - The Optimality Principle, Shortest path routing, Flooding, Distance Vector Routing, Link State Routing, Multicast routing, Routing for mobile hosts. Congestion control algorithms. Quality of Service (QoS) - requirements, Techniques for achieving good QoS.

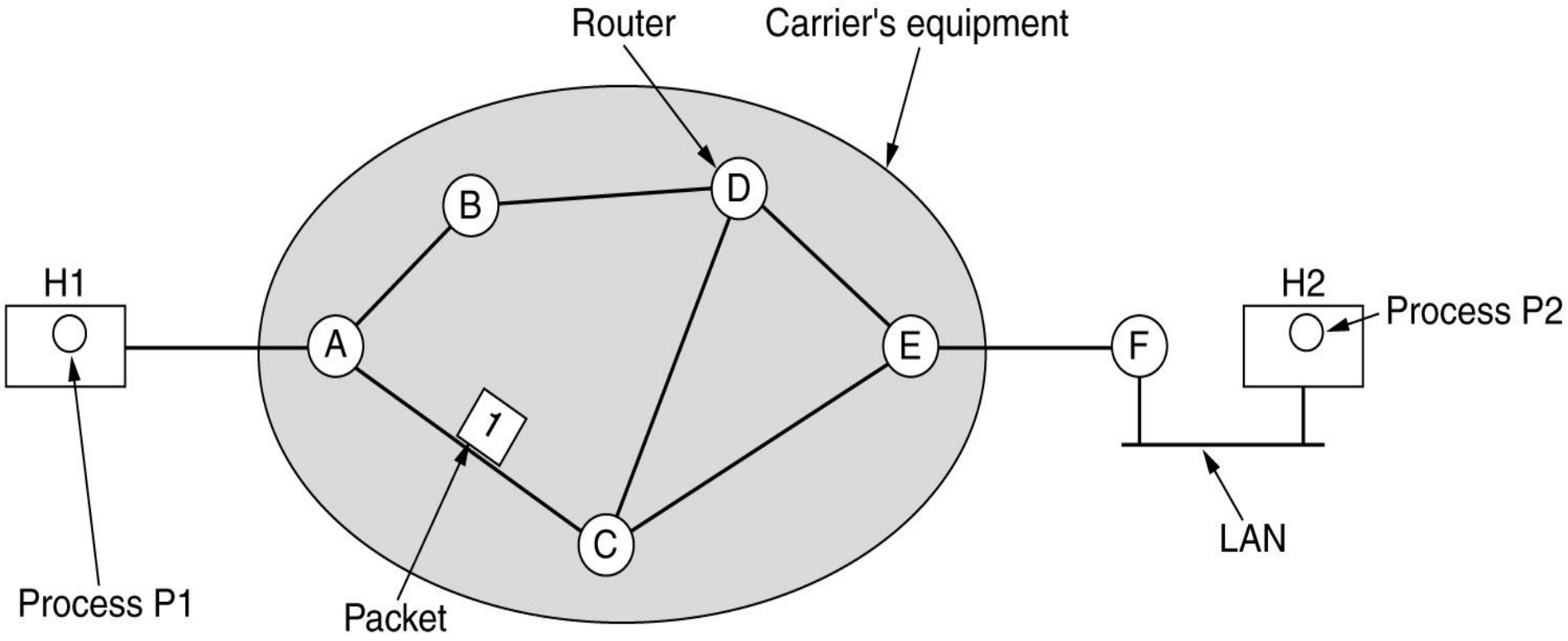
Network layer

- The network layer is responsible for host-to-host delivery and for routing the packets through the routers or switches.
- Other functions include Routing and Congestion control

Network Layer Design Issues

1. Store-and-Forward Packet Switching
2. Services Provided to the Transport Layer
3. Implementation of Connectionless Service
4. Implementation of Connection-Oriented Service
5. Comparison of Virtual-Circuit and Datagram Subnets

Store-and-Forward Packet Switching



The environment of the network layer protocols.

Store-and-Forward Packet Switching

- The major components of the system are the carrier's 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 carrier's routers, *A*, by a leased line.
- In contrast, *H2* is on a LAN with a router, *F*, owned and operated by the customer.

Store-and-Forward Packet Switching

- A host with a packet to send transmits it to the nearest router, either on its own LAN or over a point-to-point link to the carrier.
- The packet is stored there until it has fully arrived so the checksum can be verified.
- Then it is forwarded to the next router along the path until it reaches the destination host, where it is delivered.
- This mechanism is store-and-forward packet switching

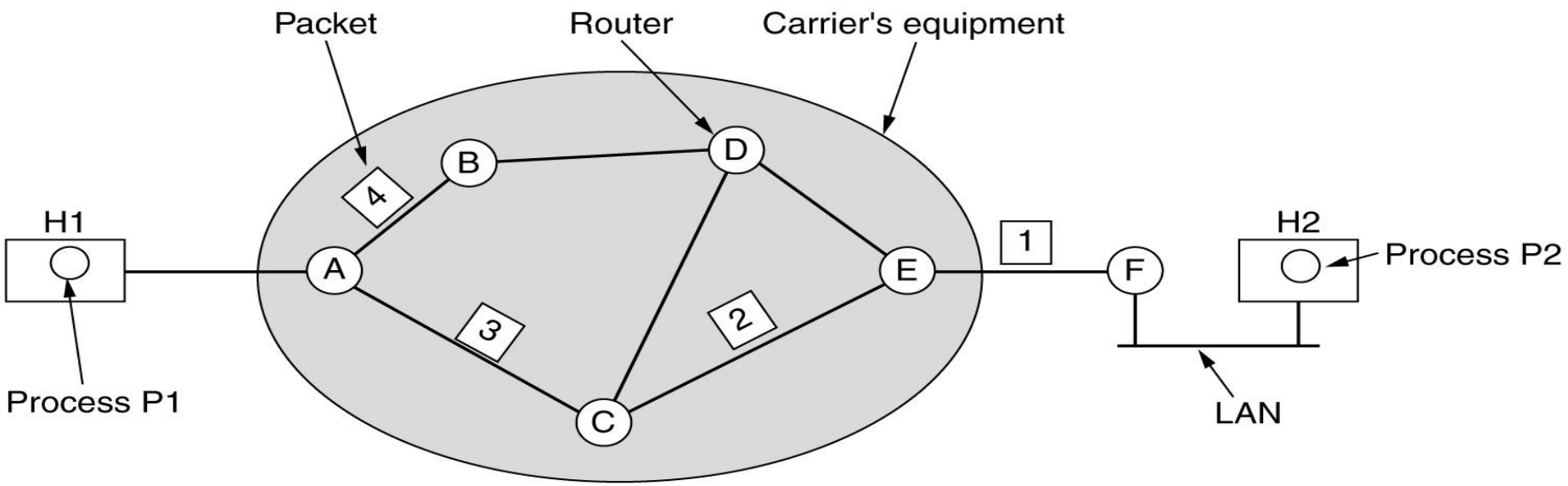
Services Provided to the Transport Layer

1. The services should be independent of the router technology.
2. The transport layer should be shielded from the number, type, and topology of the routers present.
3. The network addresses made available to the transport layer should use a uniform numbering plan, even across LANs and WANs.

Types of services by network layer

- Two different services are possible
  1. Connection Oriented
  2. Connectionless
- If connectionless service is offered, packets are injected into the subnet individually and routed independently of each other.
- No advance setup is needed.
- In this context, the packets are frequently called **datagrams** and the subnet is called a **datagram subnet**.
- If connection-oriented service is used, a path from the source router to the 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 subnet is called a **virtual-circuit subnet**.

Implementation of Connectionless Service



A's table

Dest.	initially Line	later Line
A	–	–
B	B	B
C	C	C
D	B	B
E	C	B
F	C	B

C's table

Dest.	Line
A	A
B	A
C	–
D	D
E	E
F	E

E's table

Dest.	Line
A	C
B	D
C	C
D	D
E	–
F	F

Routing within a datagram subnet.

Implementation of Connectionless Service

- Suppose that the process *P1* has a long message for *P2*
- Let us assume that the message is four times longer than the maximum packet size
- so the network layer has to break it into four packets, 1, 2, 3, and 4 and sends each of them in turn to router A using PPP.
- At this point the carrier takes over.
- Every router has an internal table telling it where to send packets for each possible destination.
- Each table entry is a pair consisting of a destination and the outgoing line to use for that destination.
- Only directly-connected lines can be used.

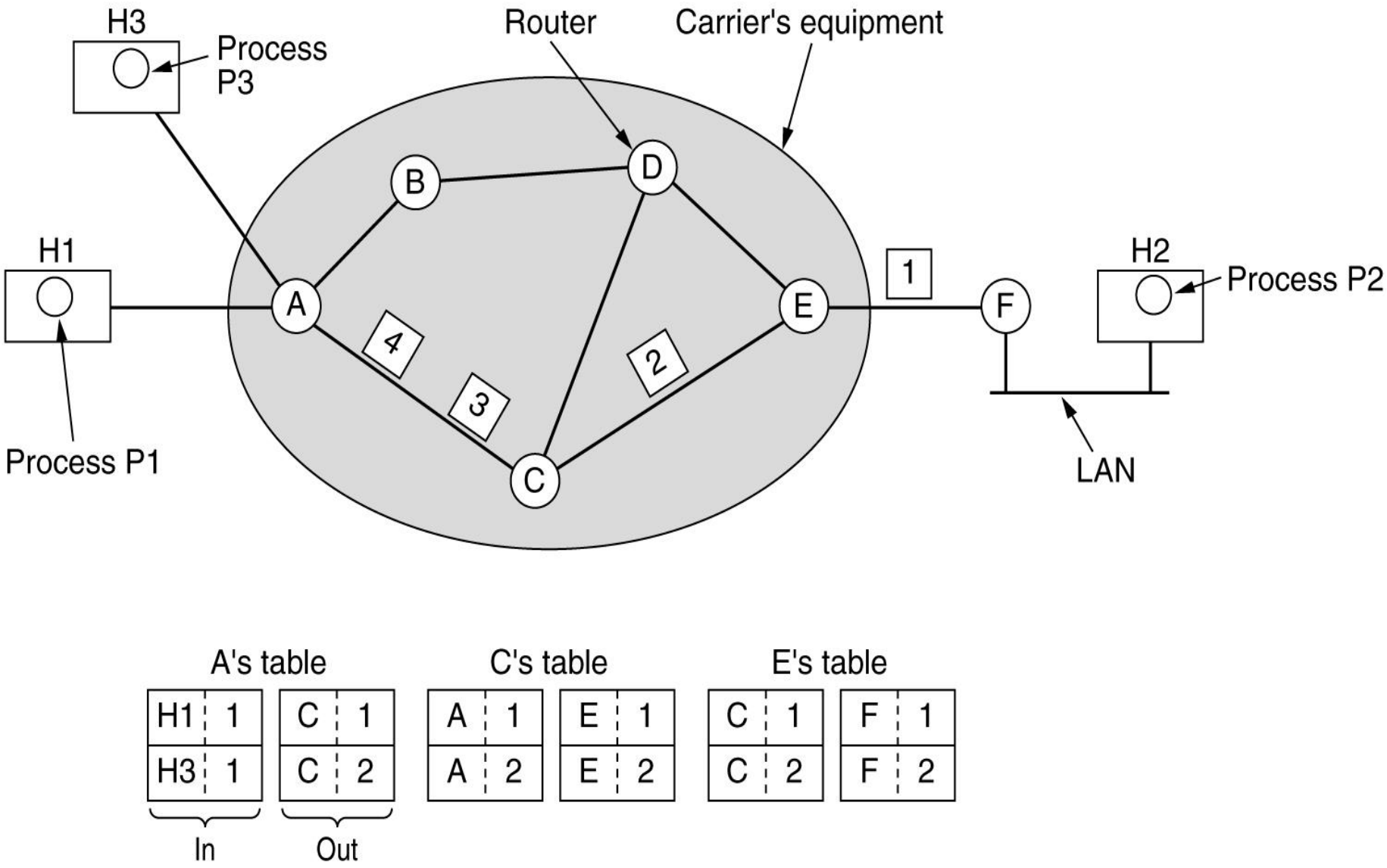
Implementation of Connectionless Service

- For example, in Fig., A has only two outgoing lines—to B and C
- so every incoming packet must be sent to one of these routers, even if the ultimate destination is some other router.
- A's initial routing table is shown in the figure under the label ‘initially’.
- As they arrived at A, packets 1, 2, and 3 were stored briefly (to verify their checksums).
- Then each was forwarded to C according to A's table.
- Packet 1 was then forwarded to E and then to F.
- When it got to F, it was encapsulated in a data link layer frame and sent to H2 over the LAN.

Implementation of Connectionless Service

- Packets 2 and 3 follow the same route
- something different happened to packet 4.
- When it got to A it was sent to router B, even though it is also destined for F.
- A learned about a traffic jam somewhere along the ACE path and updated its routing table, as shown under the label ‘later’.
- So, A decided to send packet 4 via a different route than that of the first three.
- The algorithm that manages the tables and makes the routing decisions is called the **routing algorithm**

Implementation of Connection-Oriented Service



Routing within a virtual-circuit subnet.

Implementation of Connection-Oriented Service

- For connection-oriented service, we need a virtual-circuit subnet.
- The idea behind virtual circuits is to avoid having to choose a new route for every packet sent.
- when a connection is established, a route from the source machine to the destination machine is chosen as part of the connection setup and stored in tables inside the routers.
- That 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.

Implementation of Connection-Oriented Service

- With connection-oriented service, each packet carries an identifier telling which virtual circuit it belongs to.
- Eg: In fig, host H1 has established connection 1 with host H2.
- It 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.

Implementation of Connection-Oriented Service

- 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) tells the subnet to establish the virtual circuit.
- This leads to the second row in the tables.
- Note that we have a conflict because although A can easily distinguish connection 1 packets from H1 from connection 1 packets from H3, C cannot do this.

Implementation of Connection-Oriented Service

- 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.
- In some contexts, this is called label switching

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