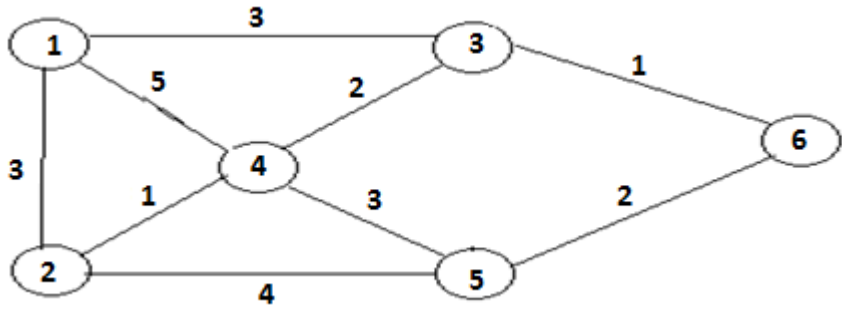


Sub:	<b>Computer Networks - 2</b>						Code:	<b>10CS64</b>	
Date:	28 / 03 / 2017	Duration:	90 mins	Max Marks:	50	Sem:	<b>VI A &amp; B</b>	Branch:	<b>ISE</b>

S.No	Answer any 5 full questions	Marks	OBE	
			CO	RBT
1.a)	Differentiate between connection oriented and connectionless service.	[05]	CO1	L2
b)	Compare the datagram packet switching and virtual packet switching.	[05]	CO1	L2
2.a)	Explain and derive delays in datagram packet switching	[07]	CO1	L2
b)	Define routing and classify the types of routing	[03]	CO2	L1
3.a)	Write short notes on hierarchical and specialized routing	[07]	CO2	L2
b)	Explain the concept of Random Early Detection (RED).	[03]	CO2	L2
4	Explain fair queuing mechanism of traffic management at packet level and also compute the expression for finish time in packet by packet fair queuing	[10]	CO2	L2
5. a)	Explain the FIFO and priority queue scheduling for managing traffic at packet level.	[06]	CO2	L2
b)	Write a note on closed loop congestion control in packet switching network.	[04]	CO2	L2
6. a)	<p>Consider the network in the following fig:</p> <p>i) Use the Dijkstra's algorithm to find the set of shortest path from node 4 to other nodes.</p> <p>ii) Find the set of associated routing table entries</p> 	[05]	CO3	L3
6.b)	Explain the Dijkstra's algorithm.	[05]	CO2	L2
7.a)	<p>Suppose we wish to transmit a large message (<math>L=10^6</math>) over three hops. Now suppose that transmission line in each hop has an error rate of <math>P=10^{-6}</math> and each hop does error checking and transmission: How many bits need to be transmitted using message switching?</p> <p>Now suppose the same above message is broken up into ten <math>10^5</math> bit packets, how many bits need to be transmitted over the three hops?</p>	[05]	CO2	L3
b)	Differentiate between the token bucket and leaky bucket for congestion control.	[05]	CO2	L2

Course Outcomes		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
<b>CO1:</b>	Differentiate between circuit switching, packet switching and message switching	-	-	-	-	-	-	-	-	-	-	-	-
<b>CO2:</b>	Implement distance vector and link state routing algorithms	-	2	-	1	-	-	-	-	-	-	-	-
<b>CO3:</b>	Select TCP/IP and UDP/IP protocols for data transmissions based on requirement	-	-	1	-	-	-	-	-	-	-	-	-
<b>CO4:</b>	Understand unicast, multicast and broadcast modes of routing and the address resolution protocols	1	-	-	-	-	-	-	-	-	-	-	-
<b>CO5:</b>	Understand different application layer protocols, data compression and security methods	1	-	-	-	-	1	-	-	-	1	-	-
<b>CO6:</b>	Understand the routing and security protocols for wireless Adhoc networks	1	-	-	-	-	-	-	-	-	-	-	-

PO1 - *Engineering knowledge*; PO2 - *Problem analysis*; PO3 - *Design/development of solutions*; PO4 - *Conduct investigations of complex problems*; PO5 - *Modern tool usage*; PO6 - *The Engineer and society*; PO7- *Environment and sustainability*; PO8 - *Ethics*; PO9 - *Individual and team work*; PO10 - *Communication*; PO11 - *Project management and finance*; PO12 - *Life-long learning*

#132, AECS Layout, IT Park Road, Kundalahalli, Bangalore 560 037  
T: +91 80 28524466/77

CMR  
INSTITUTE  
OF TECHNOLOGY



DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

**COMPUTER NETWORKS - 2**  
**IAT – 1 SOLUTION**

1. a) Differentiate between connection oriented and connectionless service.[05]

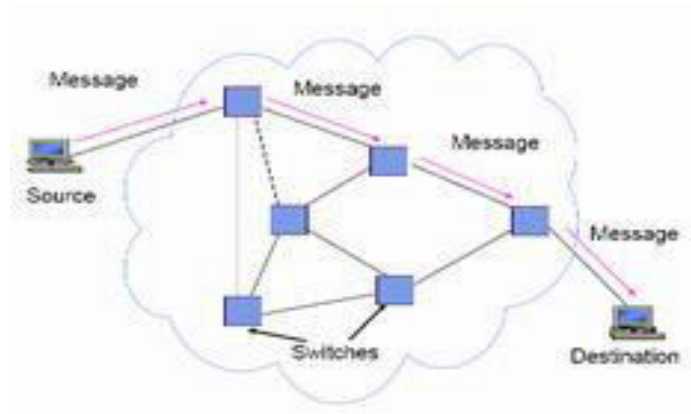
<b>Parameter</b>	<b>Connection Less Service</b>	<b>Connection Oriented Service</b>
<b>Definition</b>	It is the Communication System in which there is no need to establish virtual connection between sender and receiver.	It is the communication system in which virtual connection is established between sender and receiver before the communication begins.
<b>Data Acknowledge</b>	No data acknowledge is used, sender can not be sure about the accurate delivery of the message.	Receiver can acknowledge the data send by the sender and can re request the data if any packet fails or gets damaged.
<b>Connection Termination</b>	No Need of Connection termination.	Connection needs to be terminated after completion of communication.
<b>Packet Route</b>	Packets follow different path to reach destination and may reach in any order.	All the frames are sent through same route or path.
<b>Example</b>	Postal System	Telephone Call

1. b) Compare the datagram packet switching and virtual packet switching.[05]

<b>Datagram Packet Switching</b>	<b>Virtual Circuit Packet Switching</b>
No dedicated path	No dedicated path
Transmission of packets	Transmission of packets
Fast enough for interactive	Fast enough for interactive
Packets may be stored until delivered	Packets stored until delivered
Route established for each packet	Route established for entire conversation
Packet transmission delay	Call setup delay; packet transmission delay
Sender may be notified if packet not delivered	Sender notified of connection denial
Overload increases packet delay	Overload may block call setup; increases packet delay

2. a) Explain and derive delays in datagram packet switching [07]

### Message Switching



### Message Switching

- In message switching, a message is relayed from one switch to another until the message arrives at the destination
- A message switch operates in *store and forward* fashion (a message has to be completely received by the switch before it can be forwarded to next switch)
- At the source each message has header attached to it to provide source and destination address.
- CRC check bits are attached to detect errors
- Each switch performs an error check, and if no errors are detected, the switch examines the header to determine the next hop in the path to the destination.
- Loss of messages may occur when a switch has insufficient buffering to store the arriving message



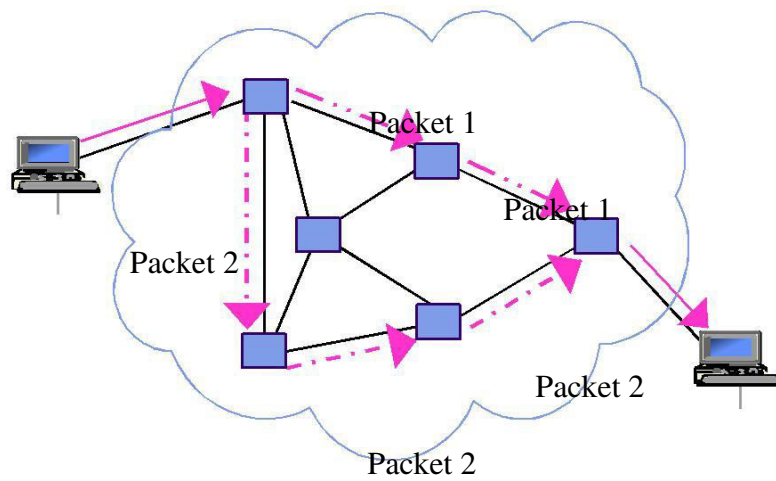
The message has to traverse the link to the first switch

- We assume
  - the link has propagation delay in seconds, T – the transmission time
- The message must traverse the link that connect two switches and from second switch to the destination.
- It follows that the minimum delay is  $3T + 3T$
- In general, the delay incurred in message switching involving L hops is  $L + LT$

### Disadvantages of message switching

- The probability of error increases with the length of the block. Thus long messages are not desirable
- Not suitable for interactive applications

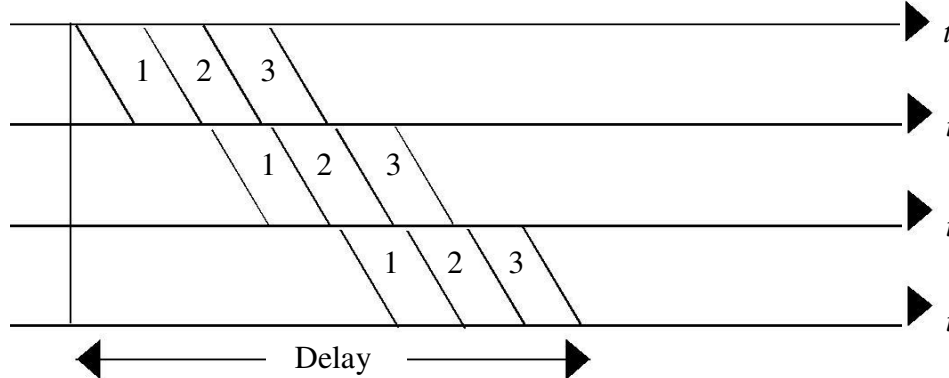
### Datagram or Connectionless Packet Switching



- Messages broken into smaller units (packets)
- Source & destination addresses in packet header
- Connectionless, packets routed independently (datagram)
- When a message arrives at the packet switch, the destination address is examined to determine the next hop.
- Packet may arrive out of order
- Re-sequencing maybe required at destination.
- Pipelining of packets across network can reduce delay, increase throughput
- Lower delay than message switching, suitable for interactive traffic

### Packet Switching Delay

Assume three packets corresponding to one message traverse same path



Minimum Delay =  $3\tau + 5(T/3)$  (single path assumed)

- Additional queuing delays possible at each link
- Packet pipelining enables message to arrive sooner

In general the delay incurred using a datagram switch involving  $L$  hops and consisting of  $k$  packets is  $L + LP + (k-1)P$

### 2. b) Define routing and classify the types of routing [03]

- 1 **Routing:** it is concerned with determining feasible path for packets to follow from each source to destination.

Classification:

- Static Vs Dynamic
- Centralized or Distributed

### 3. a) Write short notes on hierarchical and specialized routing [07]

## Hierarchical Routing

- § The hierarchical approach reduces the size of the routing tables at the routers in assigning the addresses.
- § Hosts that are near each other (i.e. a group) should have addresses that have common prefixes. The routers examine only part of the address (i.e.. the prefix) to decide how a packet should be routed.
- § Figure below gives an example of hierarchical address assignment and a flat address assignment.

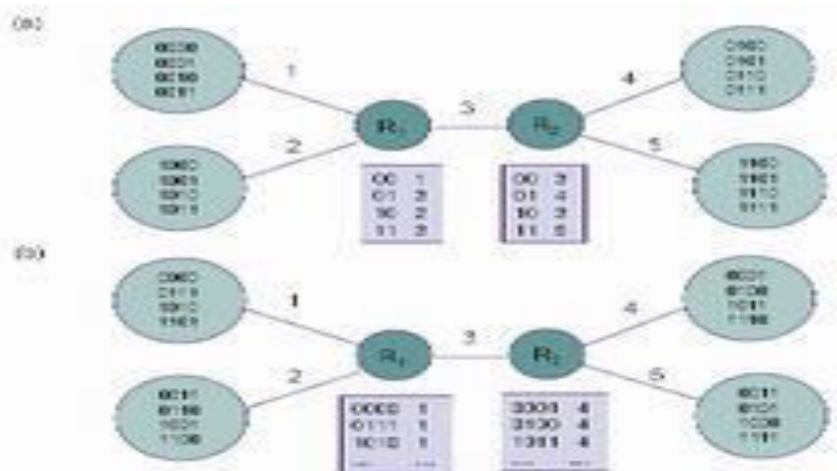
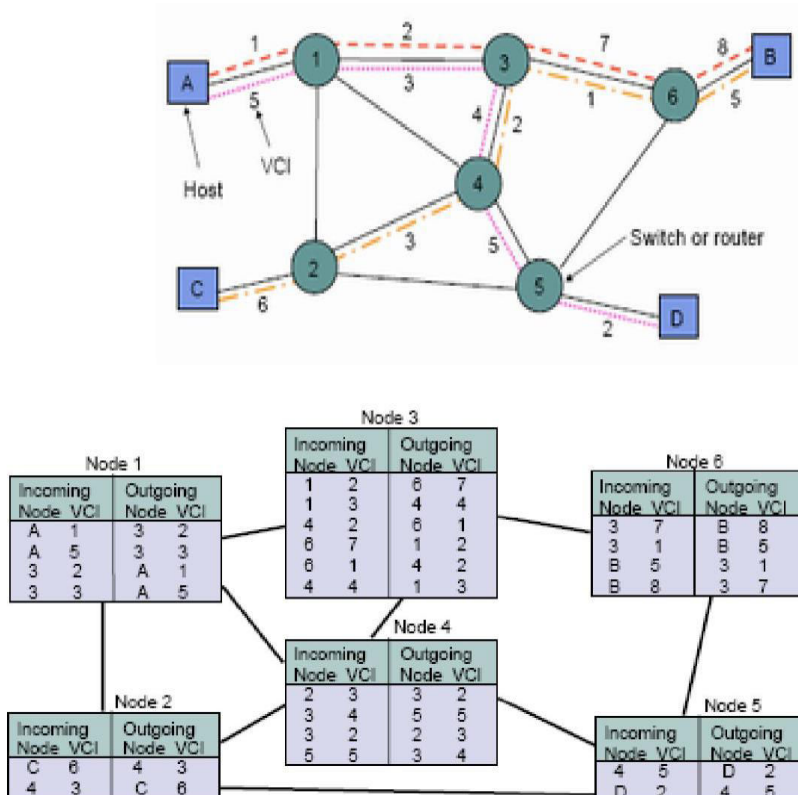


FIG (a) Hierarchical Routing and (b) Flat Routing

- § In figure (a) the hosts at each of the four sites have the same prefix. Thus the two routers need only maintain tables with four entries as shown.
- § On the other hand, if the addresses are not hierarchical (Figure 7.27b), then the routers need to maintain 16 entries in their routing tables

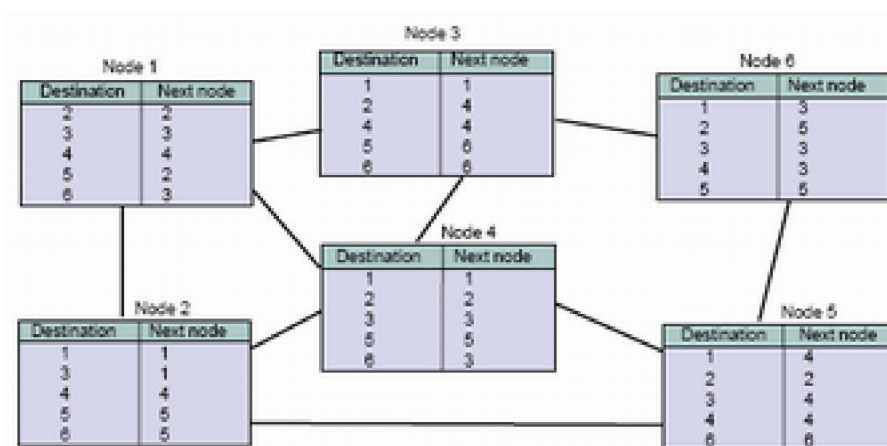
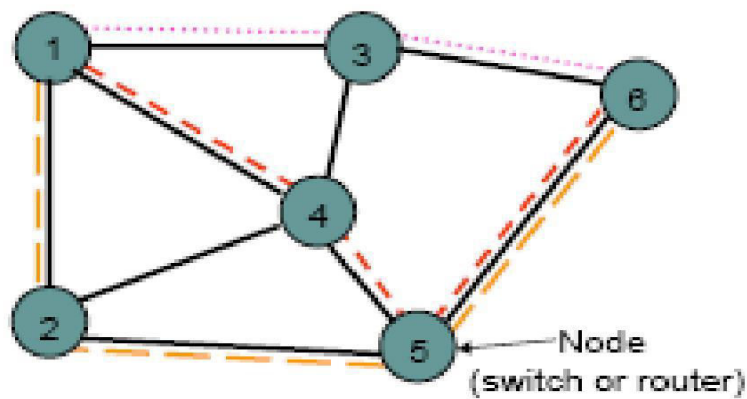
## ROUTING TABLES



## Routing Table for Virtual circuit networks

- virtual circuit identifier determines the destination

## Routing table for datagram network



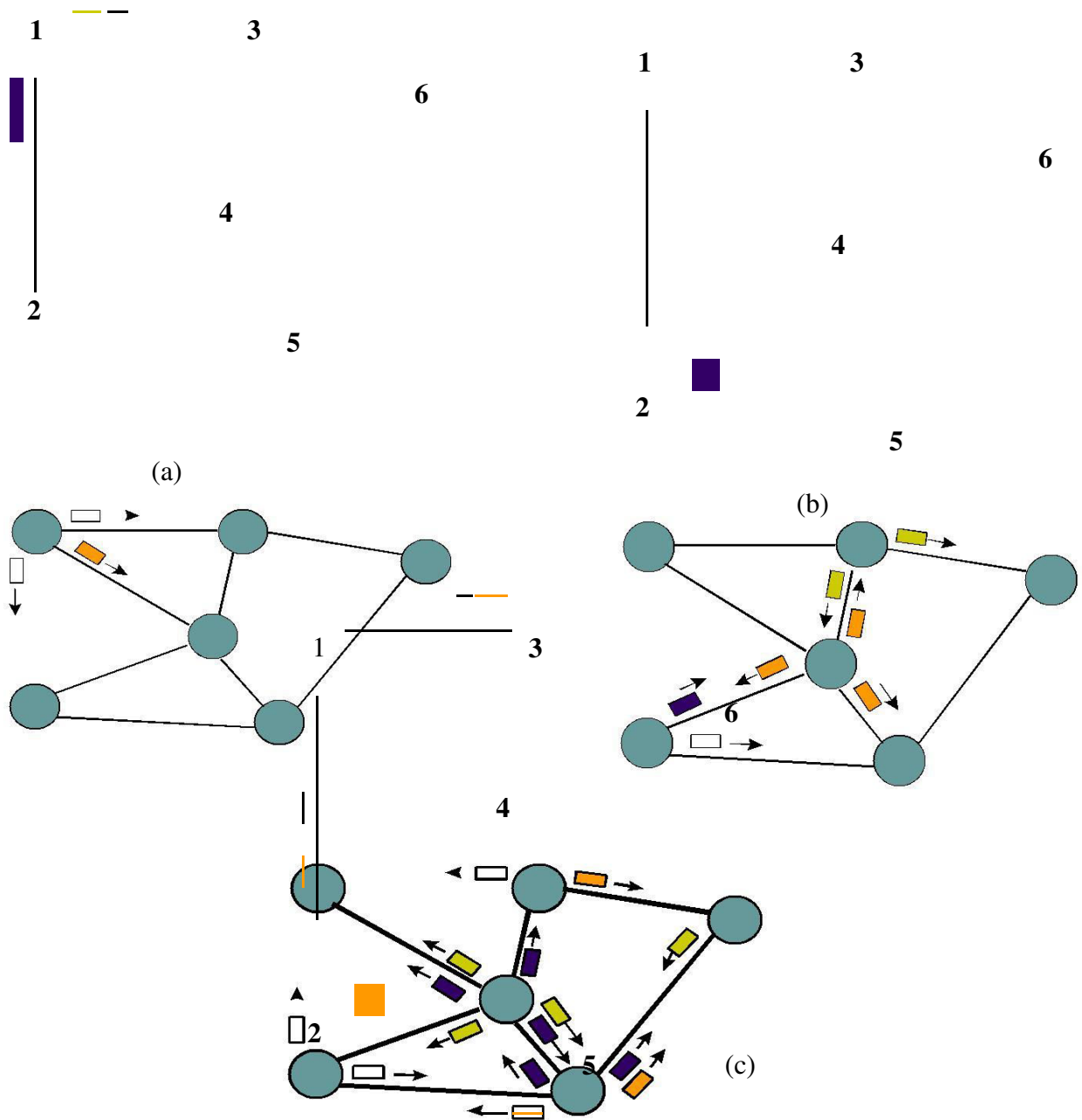
## Specialized Routing

### Flooding

- § **Principle of flooding:** a node (or a packet switch) forwards an incoming packet to all ports except to the one it arrived on.
- § Each node (a switch) performs the flooding process such that the packet will reach the destination as long as at least one path exists between the source and the destination.
- § Flooding is a useful
  - when the information in the routing tables is not available, such as during system startup,
  - when survivability is required, such as in military networks.
  - when the source needs to send a packet to all hosts connected to the network (i.e., broadcast delivery).



- Flooding generates vast numbers of duplicate packets



**Figure:** Flooding is initiated from Node 1: (a) Hop 1 transmissions (b) Hop 2 transmissions (c) Hop 3 transmissions

In figure, initially one packet arriving at node 1 triggers three packets to nodes 2, 3, and 4. In the second phase nodes 2, 3, and 4 send two, two, and three packets respectively. These packets arrive at nodes 2 through 6. In the third phase 15 more packets are generated giving a total of 25 packets after three phases.

The flooding needs to be controlled so that packets are not generated excessively.

### **How to control this?**

There are three methods to reduce the resource consumption in the network

#### **1) Use a time-to-live (TTL) field in each packet.**

- § When the source sends a packet, the time-to-live field is initially set to some small number.
- § Each node decrements the field by one before flooding the packet. If the value reaches zero, the switch discards the packet.
- § To avoid unnecessary waste of bandwidth, the time-to-live should ideally be set to the minimum hop number between two furthest nodes (called the diameter of the network).

#### **2) Add an identifier before flooding**

- § Every node adds an identifier before flooding
- § When a node identifies a packet that contains the identifier of the switch, it discards the packet.
- § This method effectively prevents a packet from going around a loop.

#### **3) Have a unique sequence number**

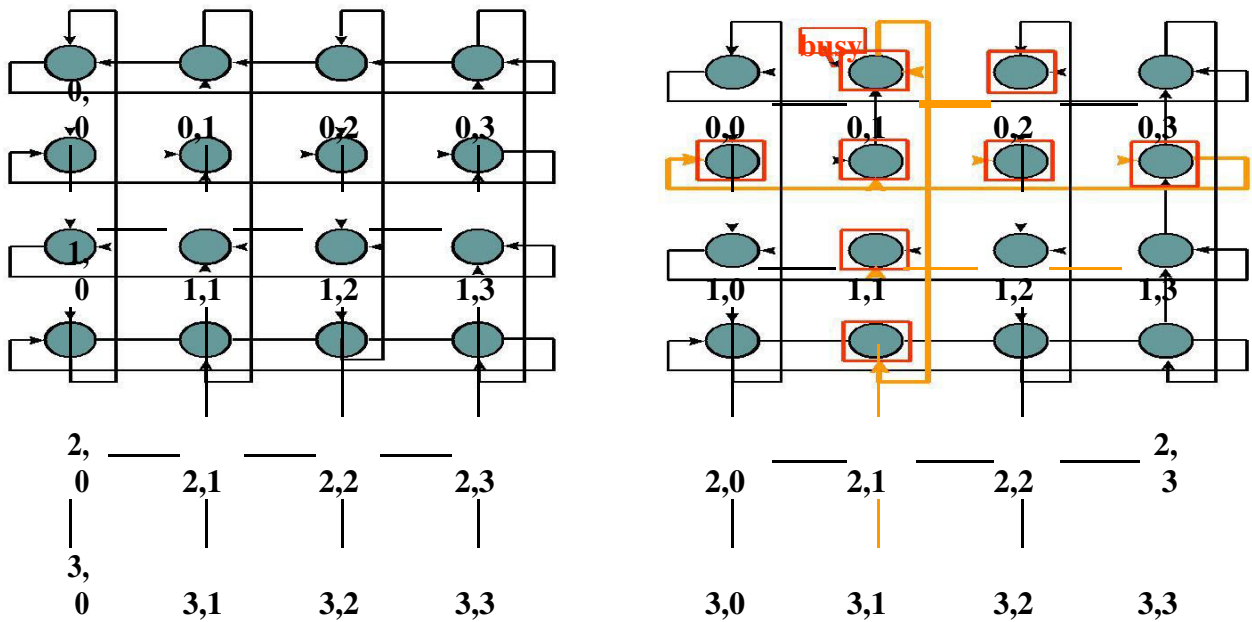
- Each packet from the given source is uniquely identified with a sequence number
- When a node receives a packet, it records the source address and the sequence number
- If node discovers that packet has already visited the node, it will discard the packet

### **Deflection Routing**

- Deflection routing was first called as Hot-potato routing.
- It requires the network to provide multiple paths for each source-destination pair.
- Each switch first tries to forward a packet to the preferred port. If the preferred port is busy or congested, the packet is deflected to another port.
- Deflection routing works well in a regular topology.

### **Example :- Manhattan Street network:**

- Each column represents an avenue, and each row represents a Street.
- Each switch is labeled (i,j) where i denotes the row number and j denotes the column number.
- The links have directions that alternate for each column or row.
- If switch (0,2) would like to send a packet to switch (1,0), the packet could go two left and one down. However, if the left port of switch (0,1) is busy (see Figure ), the packet will be deflected to switch (3,1). Then it can go through switches (2,1), (1,1), (1,2), (1,3) and eventually reach the destination switch (1,0).
- One advantage of deflection routing is that the switch can be bufferless, since packets do not have to wait for a specific port to become available. Since packets can take alternative paths, deflection routing cannot guarantee in-sequence delivery of packets.
- Deflection routing is used to implement many high-speed packet switches where the topology is very regular and high-speed buffers are relatively expensive compared to deflection routing logic.



**Figure:** Manhattan street network

**3. b) Explain the concept of Random Early Detection (RED). [03]**

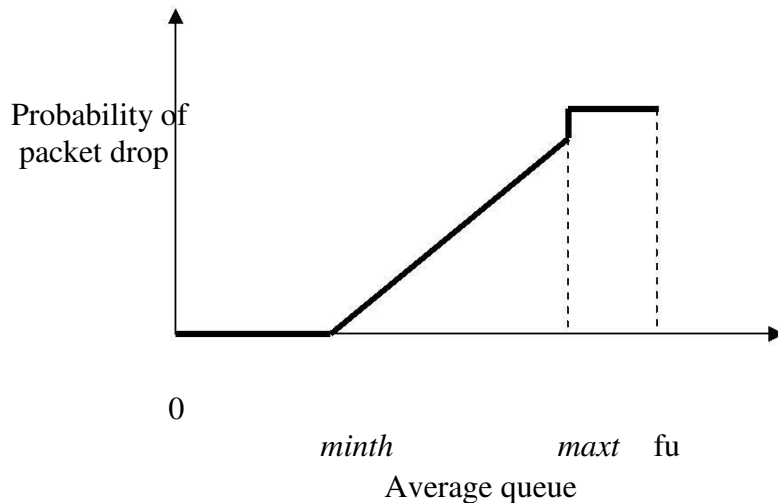
**Random Early Detection (RED)**

- An approach to preventing unfair buffer hogging by detecting congestion when a buffer begins to reach certain level and it notifies the source to reduce the rate at which they send packets.
- Packets produced by TCP will reduce input rate in response to network congestion
- RED is a buffer management technique that attempts to provide equal access to FIFO system by randomly dropping arriving packets before the buffer overflows.
- A dropped packet provides feedback information to the source and informs the source to reduce its transmission rate.
- Early drop: discard packets before buffers are full
- Random drop causes some sources to reduce rate before others, causing gradual reduction in aggregate input rate.
- $Min_{th}$  and  $Max_{th}$  are the two thresholds defined
- RED algorithm uses average queue length, when average queue length is below  $Min_{th}$ , RED does not drop any arriving packets.
- When average queue length is between  $Min_{th}$  and  $Max_{th}$ , RED drops an arriving packet with an increasing probability as the average queue length increases.
- Packet drop probability increases linearly with queue length
- RED improves performance of cooperating TCP sources.
- RED increases loss probability of misbehaving sources

**Algorithm:**

- Maintain running average of queue length
- If  $Q_{avg} < min_{threshold}$ , do nothing
- If  $Q_{avg} > max_{threshold}$ , drop packet
- If in between, drop packet according to probability

- Flows that send more packets are more likely to have packets dropped



**4. Explain fair queuing mechanism of traffic management at packet level and also compute the expression for finish time in packet by packet fair queuing. [10]**

#### **Fair Queueing / Generalized Processor Sharing**

- Fair queueing provides equal access to transmission bandwidth.
- Each user flow has its own logical queue which prevents hogging and allows differential loss probabilities
- $C$  bits/sec is allocated equally among non-empty queues.
- The transmission rate =  $C / n$  bits/second, where  $n$  is the total number of flows in the system and  $C$  is the transmission bandwidth.
- Fairness: It protects behaving sources from misbehaving sources.
- Aggregation:
  - Per-flow buffers protect flows from misbehaving flows
  - Full aggregation provides no protection
  - Aggregation into classes provided intermediate protection
- Drop priorities:
  - Drop packets from buffer according to priorities
- Maximizes network utilization & application QoS
  - Examples: layered video, policing at network edge.

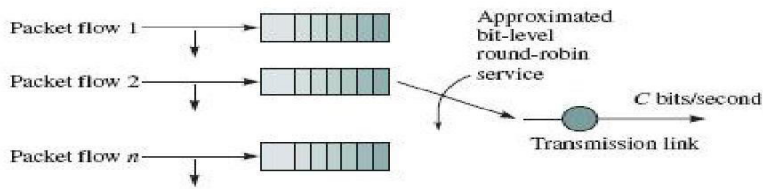
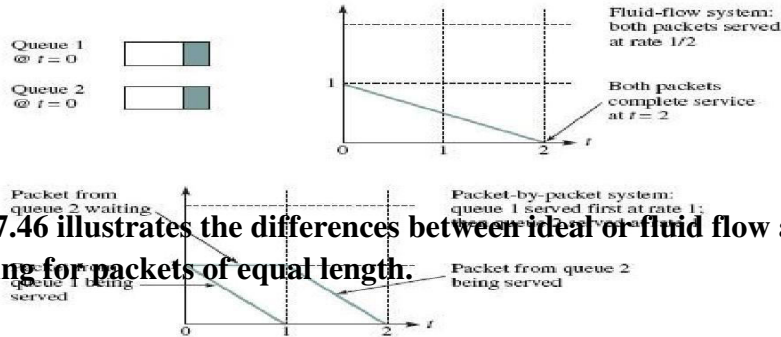


FIGURE 7.45 Fair queuing



The above figure 7.46 illustrates the differences between ideal or fluid flow and packet-by-packet fair queuing for packets of equal length.

FIGURE 7.46 Fluid-flow and packet-by-packet fair queuing (two packets of equal length)

- Idealized system assumes fluid flow from queues, where the transmission bandwidth is divided equally among all non-empty buffers.
- The figure assumes buffer 1 and buffer 2 has single  $L$ -bit packet to transmit at  $t=0$  and no subsequent packet arrive.
- Assuming capacity of  $C=L$  bits/second= $1$  packet/second.
- Fluid-flow system transmits each packet at a rate of  $1/2$  and completes the transmission of both packets exactly at time= $2$  seconds.
- Packet-by-packet fair queuing system transmits the packet from buffer 1 first and then transmits from buffer 2, so the packet completion times are  $1$  and  $2$  seconds.

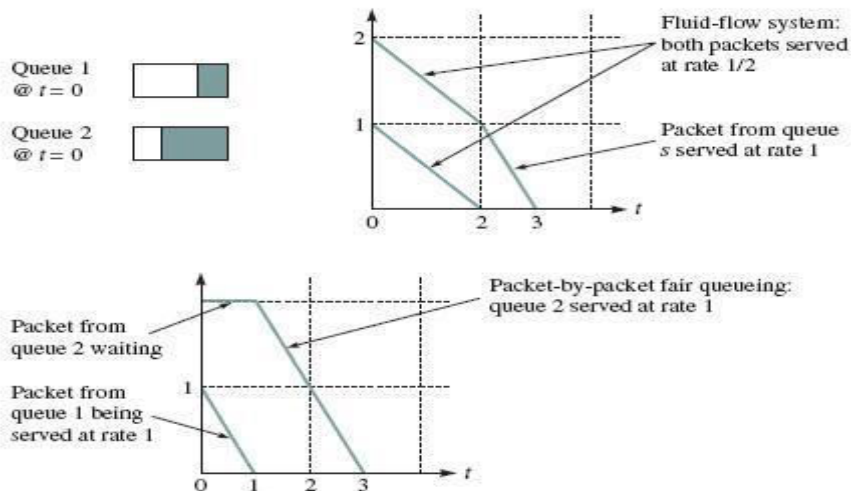
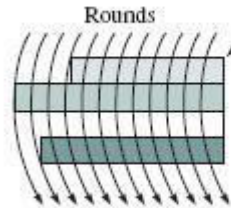


FIGURE 7.48 Fluid flow and packet-by-packet fair queuing (two packets of different lengths)

The above figure 7.48 illustrates the differences between ideal or fluid flow and packet-by-packet fair queuing for packets of variable length.

- The fluid flow fair queuing is not suitable, when packets have variable lengths.
- If the different user buffers are serviced one packet at a time in round-robin fashion, then we do not obtain fair allocation of transmission bandwidth.

- Finish tag is number used for the packet and the packet with smallest finish tag will be served first, and finish tag is computed as follows.
- Finish tag is used as priorities in packet-by-packet system.
- Consider Bit-by-Bit Fair Queueing
  - Assume  $n$  flows,  $n$  queues
  - 1 round = 1 cycle serving all  $n$  queues
  - If each queue gets 1 bit per cycle, then 1 round is the number of opportunities that each buffer has had to transmit a bit.
  - Round number = number of cycles of service that have been completed



$R(t)$  grows at rate inversely proportional to  $n_{\text{active}}(t)$

**FIGURE 7.47** Computing the finishing time in packet-by-packet fair queueing and weighted fair queueing

- If packet arrives to idle queue:
  - Finishing time = round number + packet size in bits
- If packet arrives to active queue:  
Finishing time = finishing time of last packet in queue + packet size

#### Computing the Finishing Time

$F(i, k, t)$  = finish time of  $k$ th packet that arrives at time  $t$  to flow  $i$

$P(i, k, t)$  = size of  $k$ th packet that arrives at time  $t$  to flow  $i$

$R(t)$  = round number at time  $t$

- Fair Queueing:

$$F(i, k, t) = \max \{ F(i, k-1, t), R(t) \} + P(i, k, t)$$

#### 5. a) Explain the FIFO and priority queue scheduling for managing traffic at packet level.[06]

- **FIFO QUEUEING**
- Transmission Discipline: First-In, First-Out
- All packets are transmitted in order of their arrival.
- Buffering Discipline:- Discard arriving packets if buffer is full
- Cannot provide differential QoS to different packet flows
- Difficult to determine performance delivered
- Finite buffer determines a maximum possible delay
- Buffer size determines loss probability, but depends on arrival & packet length statistics.

#### FIFO Queueing with Discard Priority

FIFO queue management can be modified to provide different characteristics of packet-loss performance to different classes of traffic.

- The above Figure 7.42 (b) shows an example with two classes of traffic.
- When number of packets in a buffer reaches a certain threshold, arrivals of lower access priority (class 2) are not allowed into the system.
- Arrivals of higher access priority (class 1) are allowed as long as the buffer is not full.

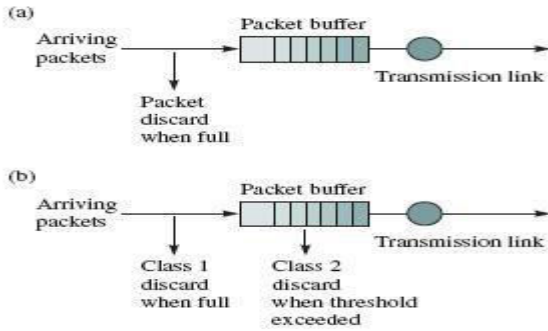


FIGURE 7.42 (a) FIFO queueing; (b) FIFO queueing with discard priority

## 2) Head of Line (HOL) Priority Queueing

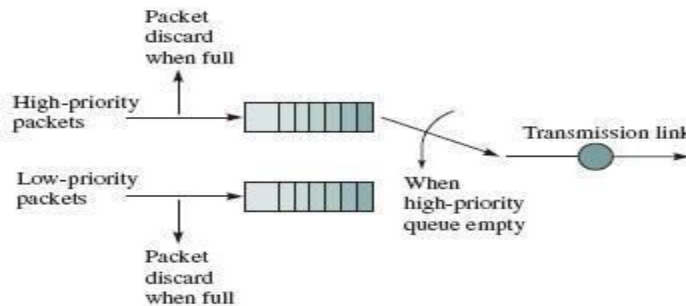


FIGURE 7.43 HOL priority queueing

- Second queue scheduling approach which defines number of priority classes.
- A separate buffer is maintained for each priority class.
- High priority queue serviced until empty and high priority queue has lower waiting time
- Buffers can be dimensioned for different loss probabilities
- Surge in high priority queue can cause low priority queue to starve for resources.
- It provides differential QoS.
- High-priority classes can hog all of the bandwidth & starve lower priority classes
- Need to provide some isolation between classes

## Sorting packets according to priority tags/Earliest due Date Scheduling

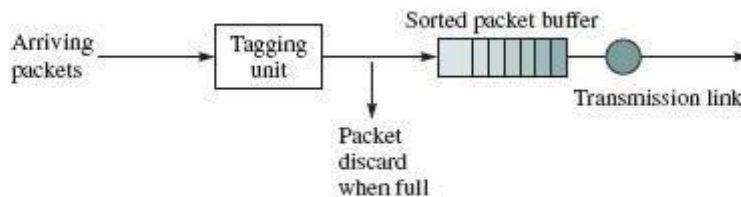


FIGURE 7.44 Sorting packets according to priority tag

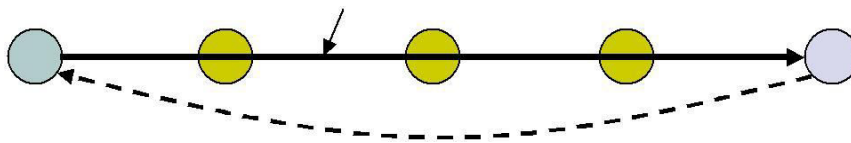
- Third approach to queue scheduling
- Sorting packets according to priority tags which reflect the urgency of packet needs to be transmitted.
- Add Priority tag to packet, which consists of priority class followed by the arrival time of a packet

- Sort the packet in queue according to tag and serve according to HOL priority system
- Queue in order of “due date”.
- The packets which requires low delay get earlier due date and packets without delay get indefinite or very long due dates

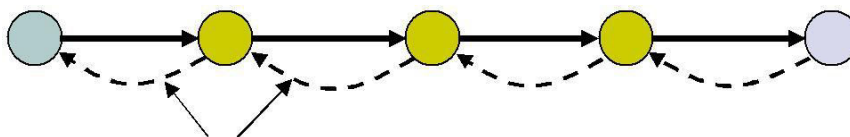
**5. b) Write a note on closed loop congestion control in packet switching network. [04]**

**Closed-Loop Flow Control**

- Congestion control
  - Feedback information is used to regulate the flow from sources into network
    - based on buffer content, link utilization, etc.
    - Examples: TCP at transport layer; congestion control at ATM level
  - Feedback information may be sent by End-to-end or Hop-by-hop.
- **End-to-end closed loop control**
  - Feedback information about state of network is propagated back to source which regulate packet flow rate.
  - Feedback information may be forwarded directly by a node that detects congestion, or it may be forwarded to destination first which then it relays information to source.
  - The transmission of feedback information introduces propagation delay, so the information may not be accurate when the source receives the information.
- **Hop-by-hop control**
  - It reacts faster than end-to-end counterpart due to shorter propagation delay.
  - State of the network is propagated to the upstream node as shown in below figure.
  - When a node detects congestion it tells to its upstream neighbor to slow down its transmission rate.
  - The Back Pressure created from one down stream node to another upstream node may continue all the way to the source.



**End-to-End vs. Hop-by-Hop Congestion Control**





**Implicit vs. Explicit Feedback:** - The information can be implicit or explicit.

**Explicit Feedback**

- The node detecting congestion initiates an explicit message to notify the source about the congestion in the network.
- The explicit message can be sent as separate packet often called as choke packets or piggybacked on a data packet.
- The explicit message may be bit information or it may contain rich amount of information.

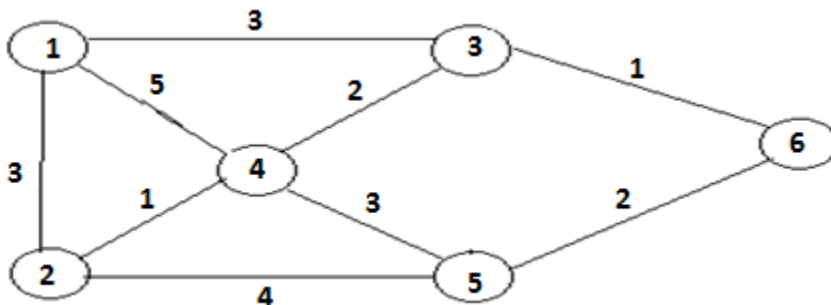
**Implicit Feedback**

- In implicit Feedback, no such explicit messages are sent between the nodes.
- Here congestion is controlled by using time out based on missing acknowledgements from destination to decide whether congestion has been encountered in the network.
- TCP congestion control is one example that regulates the transmission rate by using the implicit feedback information derived from missing acknowledgement.

6. a) Consider the network in the following fig:

i) Use the Dijkstra’s algorithm to find the set of shortest path from node 4 to other nodes.

ii) Find the set of associated routing table entries [05]



Iteration	N	D1	D2	D3	D5	D6
Initial	{4}	5	1	2	3	∞
1	{2,4}	4	1	2	3	∞
2	{2,3,4}	4	1	2	3	3
3	{2,3,4,5}	4	1	2	3	3
4	{2,3,4,5,6}	4	1	2	3	3

5	{1,2,3,4,5,6}	4	1	2	3	3

**Datagram routing table**

Node	Next Node	Cost
1	4	4
2	4	1
3	4	2
5	4	3
6	3	3

**6. b) Explain the Dijkstra's algorithm.[05]**

**Dijkstra Algorithm:**

**Finding shortest paths in order**

- $N$ : set of nodes for which shortest path already found
- Initialization: (Start with source node  $s$ )
- $N = \{s\}$ ,  $D_s = 0$ , " $s$  is distance zero from itself"
- $D_j = C_{sj}$  for all  $j \neq s$ , distances of directly-connected neighbors
- Step A: (Find next closest node  $i$ )
- Find  $i \notin N$  such that
  - $D_i = \min_{j \notin N} D_j$
- Add  $i$  to  $N$
- If  $N$  contains all the nodes, stop
- Step B: (update minimum costs)
- For each node  $j \notin N$
- $D_j = \min(D_j, D_i + C_{ij})$
- Go to Step A

**7. a) Suppose we wish to transmit a large message ( $L=10^6$ ) over three hops. Now suppose that transmission line in each hop has an error rate of  $P=10^{-6}$  and each hop does error checking and transmission: How many bits need to be transmitted using message switching? Now suppose the same above message is broken up into ten  $10^5$  bit packets, how many bits need to be transmitted over the three hops?[05]**

Soln.      given       $p = 10^{-6}$   
 $L = 10^6 = 1000000$

probability that the message arrives correctly after the first hop is,

$$P_c = (1 - p)^L$$

$$= (1 - 10^{-6})^{1000000} \approx e^{-LP} = e^{-1} = \frac{1}{e} \approx \frac{1}{2.718}$$

i.e.) On the average it will take three tries to get the message over the first hop.

So, second hop will require another 3 full msg trans

$\therefore$  Total of 6 Mbits across 2 hops.

$\rightarrow$  If msg is broken into ten  $10^5$  bit packets, The probability after first hop is,

$$P_c' = (1 - 10^{-6})^{100000} \approx e^{-1/10} \approx 0.90$$

$\therefore$  each packet needs to be transmitted  $= \frac{1}{0.90} = 1.1$  times for 2 hops = 2.2 Mbits.

7. b) Differentiate between the token bucket and leaky bucket for congestion control.[05]

S.No	LEAKY BUCKET	TOKEN BUCKET
1	It sends the packet at constant rate	It allows large bursts to be sent faster rate after that constant rate.
2	If bucket is full packets are discarded(no concept of queue)	If bucket is full token are discarded, but not the packet. (Infinite queue).
3	Packets are transmitted continuously	Packets are transmitted when there are enough token
4	Token independent	Token dependent
5	Smooth out traffic by passing packets only when there is a token. Does not permit burstiness.	Token bucket smooths traffic too but permits burstiness - which is equivalent to the number of tokens accumulated in the bucket
6	Application: Traffic shaping or traffic policing.	Application: Network traffic shaping or rate limiting.