MONASH University
Information Technology

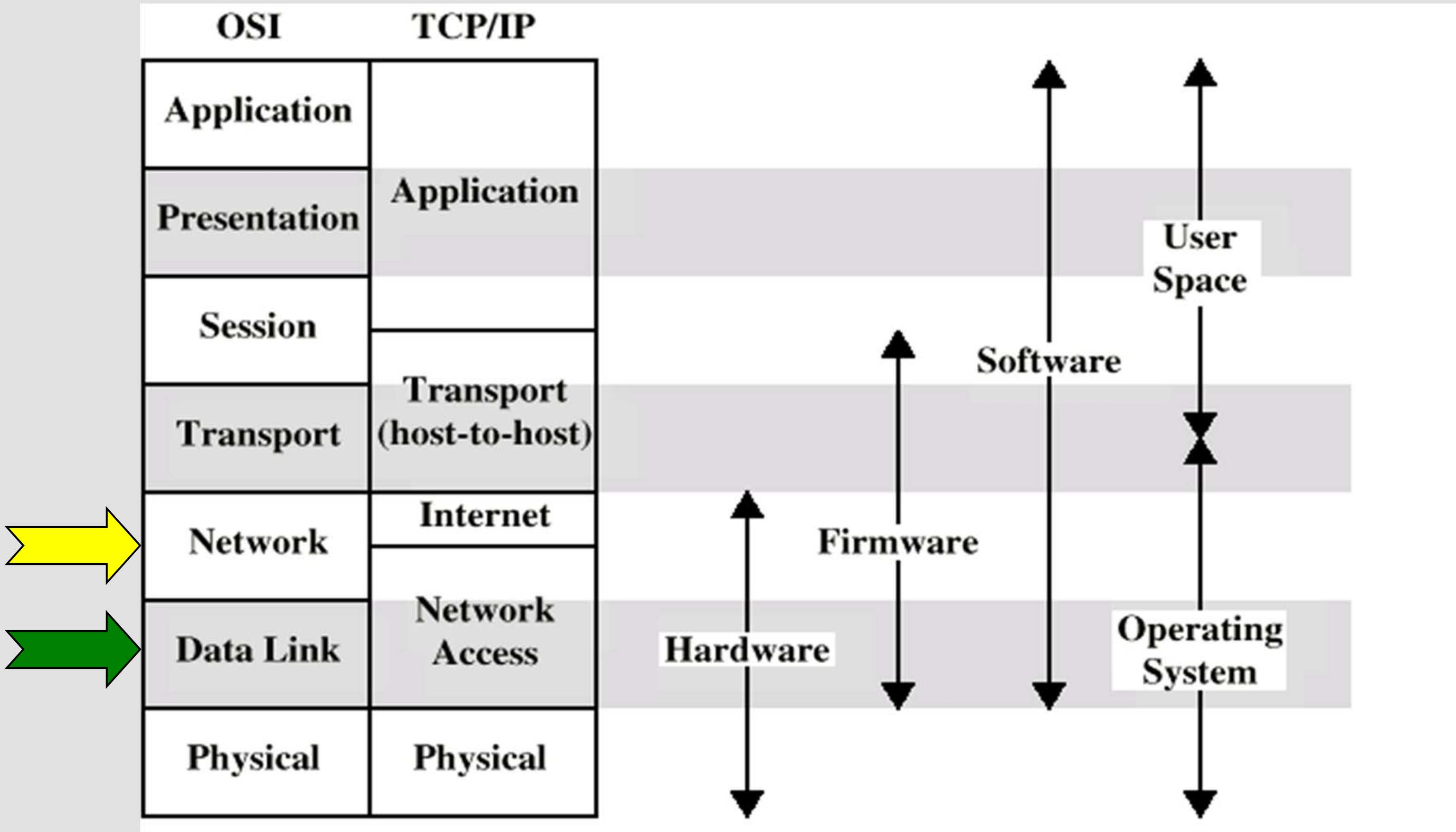
# FIT2020 - Network Architecture

## Lecture 7: Network Layer

# Last Lecture

- **Overview of OSI model**
- **Protocols theoretical know how**
- **Data link layer overview**
- **TCPIP overview**
- **Design issues**

# OSI and TCPIP



# This Lecture

- **What is Network Layer**
- **Role of Network layer**
- **Connection-oriented/connectionless services**
- **Routing ?**
- **Routing algorithms**
- **Congestion control**

# Network Layer

- **This layer deal with end to end communication**
  - Transfer of packets from source to destination
  - The lowest layer to deal with end to end transmission
- **Data link and Physical layers transfer frames between nodes directly connected by transmission medium**
  - Transport layer
    - Error control
    - Flow control

# Conti …

- **Network layer functions**
  - Routing
    - > Finding path from source to destination
  - Congestion control
    - > Prevention of service degradation from heavy traffic
  - Internetworking
    - > Connecting small networks to form a big network
- **Need to know the topology of the network e.g. set of routers and appropriate path to destination**

# Network Layer Services

- **Network layer is owned by network provider – no user control**
- **Network layer provides service to transport layer**
- **Transport layer is owned by user**
- **Transport layer**
  - Port number concept
  - Should not be concerned about number, type and topology of subnets

# Conti …

- Transport and network layer interface is between carrier and customer
- Network addresses provided to transport layers should use uniform numbering plan across LANs and WANs
- There are big concerns to users about the reliability and error detection of the transmitted data

# Conti …

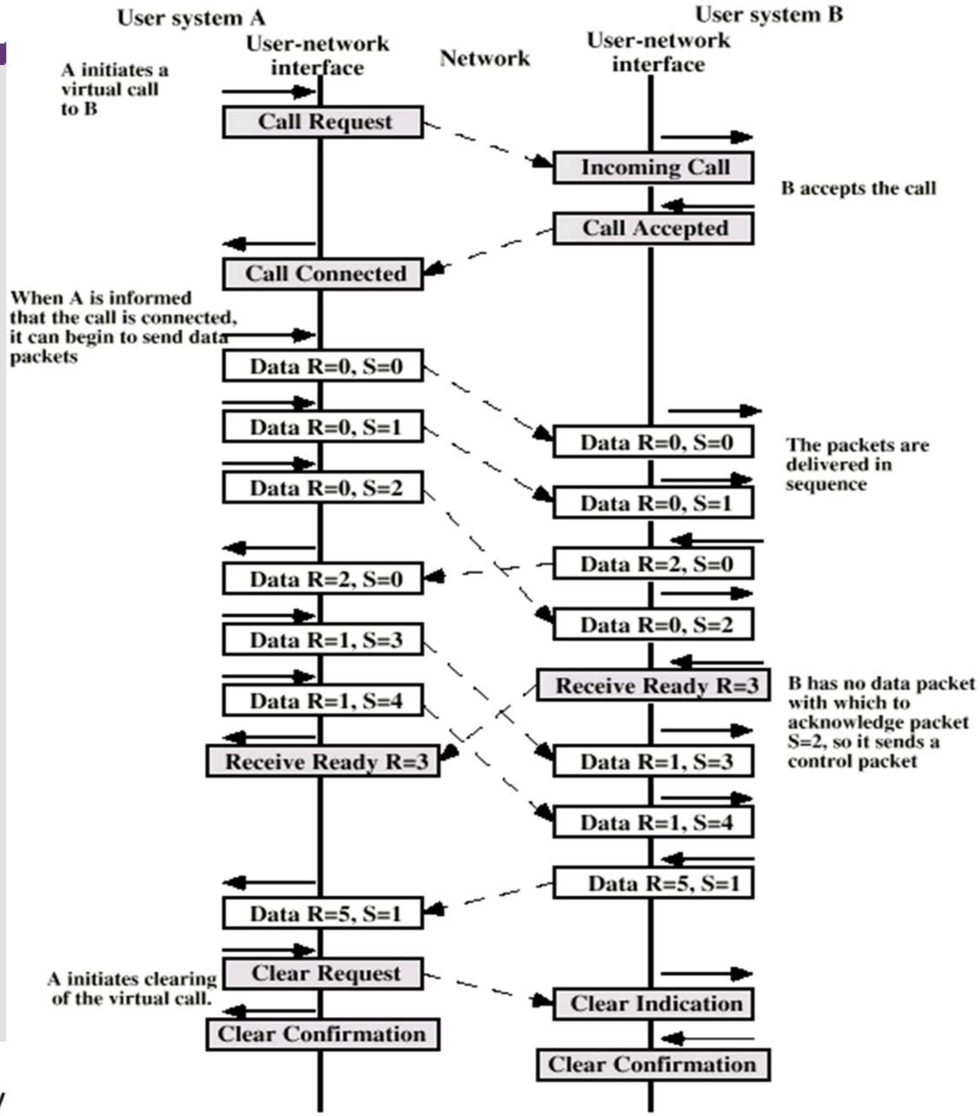
- Network layer provides two services
  - Connection oriented
  - Connection less

- Transport layer user does bear the impact of the service selection
- Service selection may impact the amount to be paid for service

# Connection-Oriented Service

- Connection has to be established before any data can be transmitted - telephone
- Network-process (on sending side) must set up a connection to it's peer on receiving side
- This connection should be identified by special identifier: virtual circuit numbers (sending and receiving)
- These identifier remain fixed during the whole session

# Virtual Call



# Connection-Oriented Service

- During connection set up, capabilities can be negotiated e.g. parameters, quality and cost of service
- Packets follow same path and delivered in sequence
- Communication in both direction

# Conti …

- **Guaranteed delivery**
- **Explicit confirmation of delivery**
  - Reliable service
- **Priority can be set for fast delivery of packets**
- **In this mode of communication, complexity is in network layer**
- **Services: Real time audio and video**
- **Example: ATM network**

# Connectionless Service

- No connection is needed
- Packets are transmitted by the selection of independent paths
- Routing information in the header of each packet
- Packets may be lost on the way, or they may arrive at the destination in the different order as they are transmitted.
- It is up to the transport layer to do the error control.
- The packets transmitted this way are sometime called datagrams, in analogy with the telegrams.

# Conti …

- Routers do not have a table with one entry for each open virtual circuit. Instead, they have a table telling which outgoing line to use; based on destination addresses.
- In this mode of communication, complexity is in transport layer
- Services: Real time data, digitize voice (speedy delivery)
- Example: Internet

# Connection-oriented verses Connectionless

- **Two issues**
  - Connection-oriented or Connectionless
  - Reliable or unreliable
  - Possible four combination
  - Only two combination are mainly used:
    - > Connection-oriented and reliable
    - > Connectionless and unreliable
  - Connection-oriented is called virtual circuit, while connection-less is called datagram

# Conti …

- **Connection-oriented (ATM)**
  - > Establishment of virtual circuit
  - > Routers maintain detailed-table for each virtual connection
  - > Packets (VCN+Sequence number+etc) are routed on pre-known routing information
- **Connectionless (Internet)**
  - > Each packet contain routing information
  - > Routers maintain table for outgoing lines for each possible router

# Conti …

- **Internet community verses telephone companies**
  - Internet community (transport layer) does not want to trust Network layer
    - > Connectionless mode with primitives
      - Send packets
      - Receive packets
  - Telephone companies believe that connection-oriented service (reliable) should be provided by Network layer

# Trade off

  - Trade-off between router memory (CO) space and bandwidth (CL).
  - Trade-off between connection setup time and address parsing time.
- **ATM-TCPIP (Internet)**
    - ATM Network layer establishes connection-oriented link
    - Independent IP packets on established link
    - Inefficient approach as same functionality is doubled
      - ATM network layer guarantees ordered packet deliver
      - TCP tries to order the delivered packets again

# Services and Subnet

- Connectionless (Robust, Easy to fail, congestion)
  - Datagram
    - UDP over IP
  - Virtual Circuit
    - UDP over IP over ATM
- Connection-oriented
  - Datagram
    - TCP over IP
  - Virtual Circuit
    - ATM AAL1over ATM

# Network Layer Functions

- **Routing**
  - Many paths are possible from source to destination
  - Selected path should cause least transmission delay and offer high throughput
  - Routing will be different for both mode of communication:
    - Connection-oriented
    - Connectionless

# Conti …

- **Complex, crucial aspect of packet switched networks: routing**
- **Characteristics requirements:**
  - Correctness
  - Simplicity
  - Robustness – delivery in localized failure
  - Stability – competing with robustness
  - Trade off between Fairness and Optimality
    - > For congestion optimality may undermine fairness
  - Efficiency

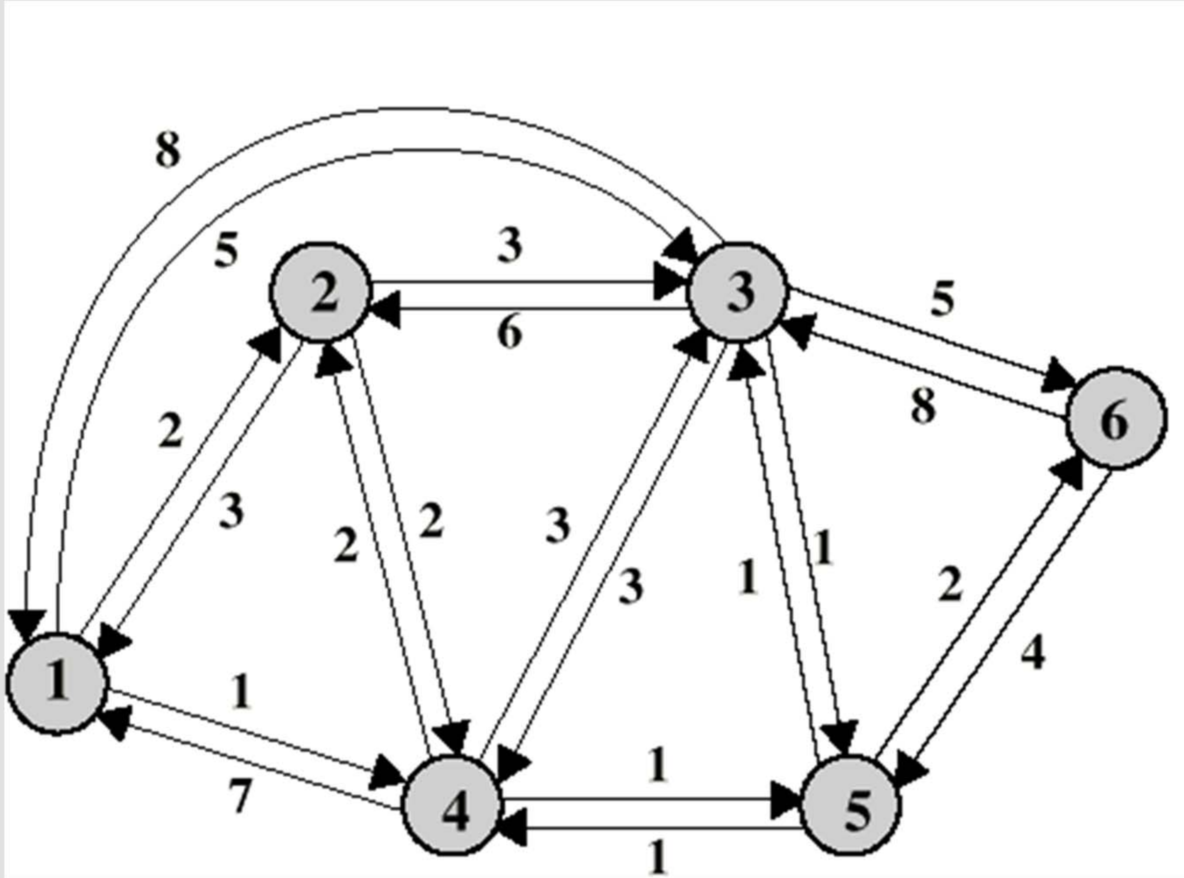
# Requirements and Design Elements

- Performance Criteria
- Decision Time and Place for Routing
- Network Information Source and Update Timing

# Performance Criteria (Design Criteria)

- The performance is normally measured using some criteria
  - **Minimum hop**
    - > **one hop is one node-to-node link**
  - **Packet delay**
  - **Fairness**
  - **Total network throughput. Based on these criteria, each link is assigned a cost value.**
  - **Least cost**
    - > **The total cost of a path between the source and the destination is defined as the sum of the costs of the links traversed.**
    - > **The routing algorithm is to find the path with least cost.**

# Least Cost



# Conti …

- **Cost could be associated with data rate**
  - Cost inversely proportional to data rate (High data rate – low cost)
  - Least cost should provide highest throughput
- **Cost could be associated with queue delay**
  - Least cost should provide low delay
- **From where these cost information should come**
- **There is jungle of router out there**

# Decision Time and Place for Routing

- Time
  - For datagram operation router decision for each packet
  - Virtual circuit basis – at start of session establishment
- Place
  - Distributed
    - > Made by each node
    - > Most common
  - Centralized – nominated node
    - > Loss of this node may hurt the operation of the network
  - Source – user may indicate routing information

# Network Information Source and Update Timing

- Routing decisions usually based on knowledge of network (not always)
- Distributed routing
  - > Nodes use local knowledge
  - > May collect info from adjacent nodes
  - > May collect info from all nodes on a potential route
- Central routing
  - > Collect info from all nodes
- Update timing
  - > When network info is held by nodes then updated
    - – **Fixed - never updated**
    - – **Adaptive - regular updates**

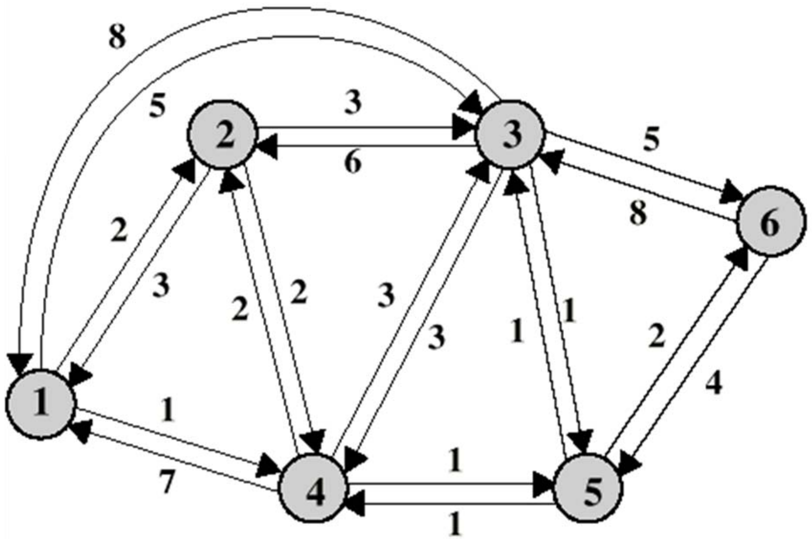
# Routing Strategies

- Fixed
- Flooding
- Random
- Adaptive

# Fixed Routing

- **Single permanent route for each source to destination pair**
- **Determine routes using a least cost algorithm**
  - Dijkstra's Algorithm
  - Bellman-Ford Algorithm
- **Route fixed, at least until a change in network topology**
- **No flexibility – does not respond to network congestion or failure**

# Fixed Route Table




**CENTRAL ROUTING DIRECTORY**

| To Node \ From Node | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | — | 1 | 5 | 2 | 4 | 5 |
| 2 | 2 | — | 5 | 2 | 4 | 5 |
| 3 | 4 | 3 | — | 5 | 3 | 5 |
| 4 | 4 | 4 | 5 | — | 4 | 5 |
| 5 | 4 | 4 | 5 | 5 | — | 5 |
| 6 | 4 | 4 | 5 | 5 | 6 | — |

**Node 1 Directory**

| Destination | Next Node |
|---|---|
| 2 | 2 |
| 3 | 4 |
| 4 | 4 |
| 5 | 4 |
| 6 | 4 |

**Node 2 Directory**

| Destination | Next Node |
|---|---|
| 1 | 1 |
| 3 | 3 |
| 4 | 4 |
| 5 | 4 |
| 6 | 4 |

**Node 3 Directory**

| Destination | Next Node |
|---|---|
| 1 | 5 |
| 2 | 5 |
| 4 | 5 |
| 5 | 5 |
| 6 | 5 |

**Node 4 Directory**

| Destination | Next Node |
|---|---|
| 1 | 2 |
| 2 | 2 |
| 3 | 5 |
| 5 | 5 |
| 6 | 5 |

**Node 5 Directory**

| Destination | Next Node |
|---|---|
| 1 | 4 |
| 2 | 4 |
| 3 | 3 |
| 4 | 4 |
| 6 | 6 |

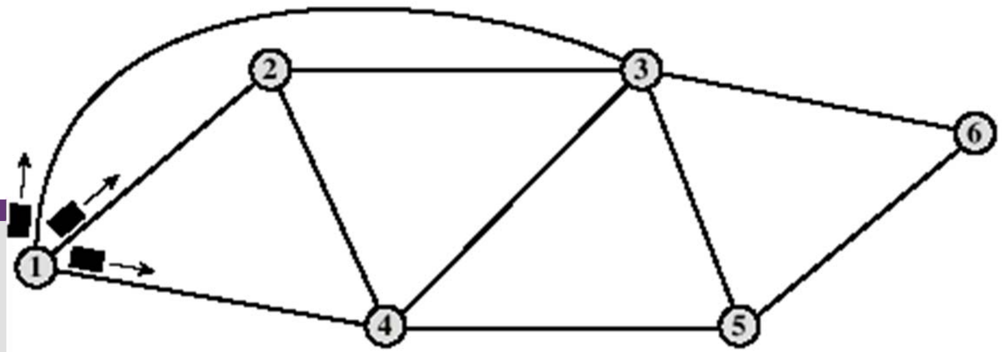
**Node 6 Directory**

| Destination | Next Node |
|---|---|
| 1 | 5 |
| 2 | 5 |
| 3 | 5 |
| 4 | 5 |
| 5 | 5 |

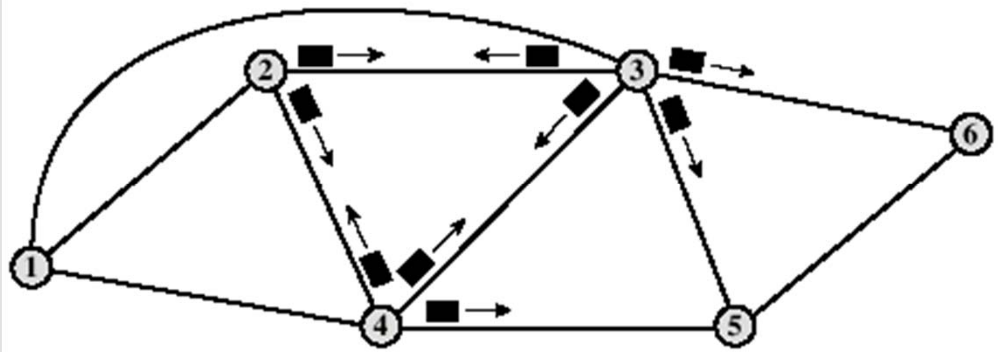
# Flooding

- No network info required
- Packet sent by node to every neighbor
- Incoming packets retransmitted on every link except incoming link
- Eventually a number of copies will arrive at destination
- Each packet is uniquely numbered so duplicates can be discarded
- Nodes can remember packets already forwarded to keep network load in bounds
- Can include a hop count in packets

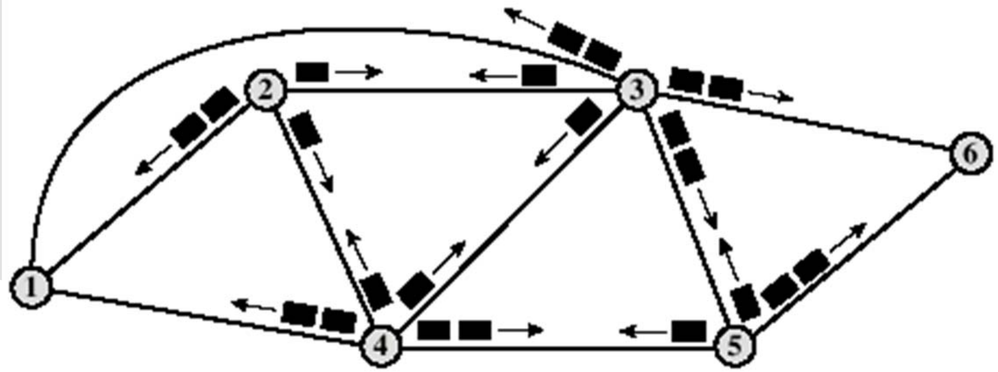
# Flooding




(a) First hop




(b) Second hop




(c) Third hop

MONASH University
Information Technology

# Properties of Flooding

- **All possible routes are tried**
  - Very robust
- **At least one packet will have taken minimum hop count route**
  - Can be used to set up virtual circuit
- **All nodes are visited**
  - Useful to distribute information (e.g. routing)
- **Emergency messages and military messages**

# Random Routing

- Random routing has simplicity and robustness of flooding but less traffic
- Node selects one outgoing path for retransmission of incoming packet
  - Selection can be random or round robin excluding the link on which the packet arrived
- Can select outgoing path based on probability calculation depending on data rate
  - $P_i = R_i / \sum_j (R_j)$
- No network info needed
- Route is typically not least cost nor minimum hop

# Adaptive Routing

- **Some times it is called dynamic routing**
- **Used by almost all packet switching networks**
- **Routing decisions change as conditions on the network change**
  - Failure
  - Congestion
- **Requires info about network**
- **Decisions more complex**
  - Tradeoff between quality of network info and overhead
- **Reaction too quickly can cause oscillation**
  - Too slowly to be irrelevant

# Adaptive Routing - Advantages

- **Improved performance**
- **Aid congestion control**
- **Complex system**
  - May not realize theoretical benefits

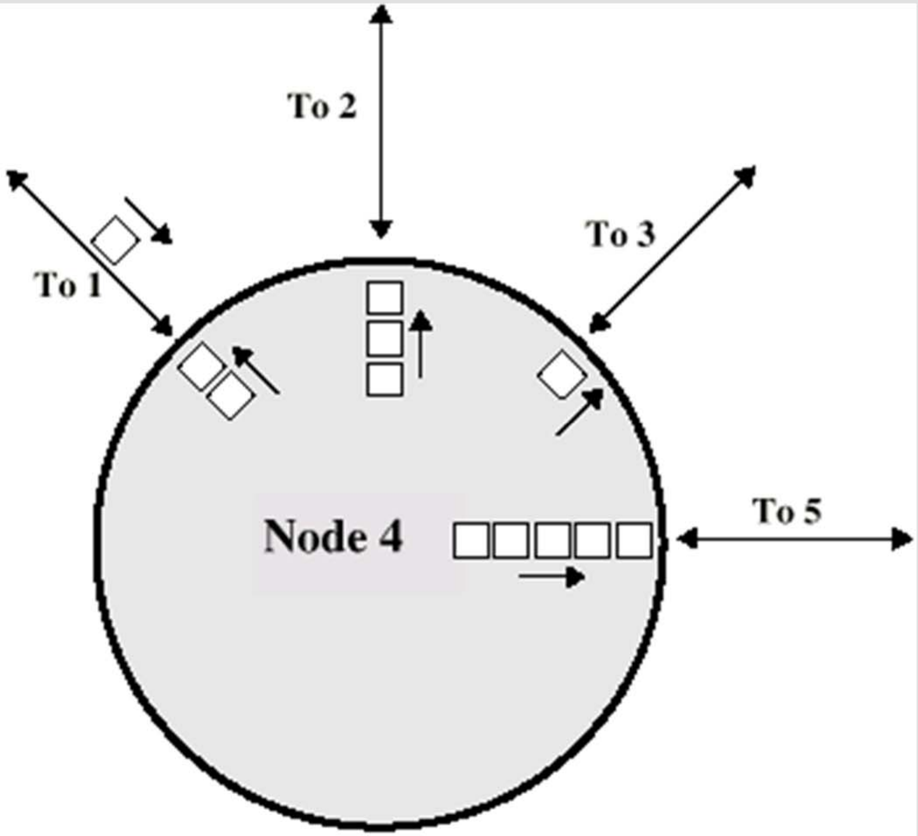
# Isolated Adaptive Routing

- **Each node monitors in coming and out going queues**
- **Collects traffic statistics of out going lines**
- **If there is significant change, it may decide to change the route for following packets**
  - Shorter out going length could be given preference
  - But shorter queue could lead to wrong direction
    - > Bias is added to prevent travelling in the wrong direction

# Isolated Adaptive Routing =>(Q+B)

Node 4's Bias
Table for
Destination 6

| Next Node | Bias |
|---|---|
| 1 | 9 |
| 2 | 6 |
| 3 | 3 |
| 5 | 0 |



# Distributed Adaptive Routing

- Each node calculates it’s own routing table
- Each node periodically transmits its status to it’s neighbors
- This information goes across the entire network from node to node and they update their routing tables accordingly

# Centralised Adaptive Routing

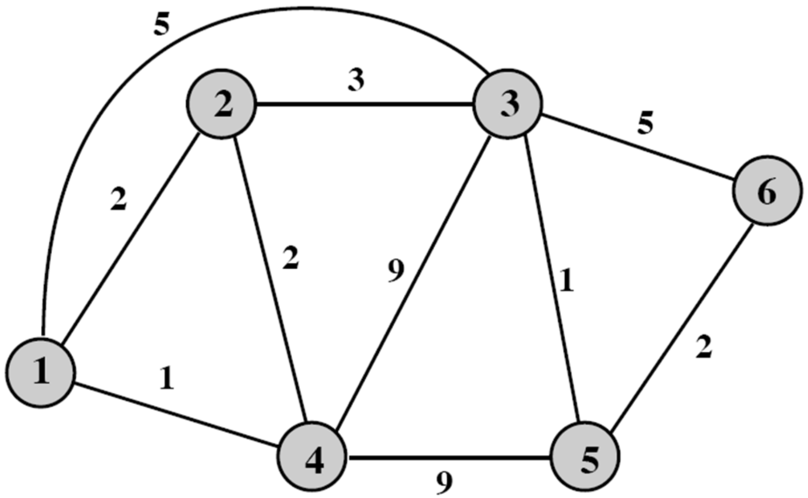
- One node is network routing manager
- All nodes periodically forward such status information:
  - Queue lengths on outgoing and incoming lines
  - Number of packets processed within the most recent interval.
  - The routing manager thereby is provided with an overview of network functioning, where bottlenecks are occurring, and where facilities are under-utilized etc.
  - The routing manager periodically recalculates the optimal paths between nodes and constructs and distributes new routing tables to all nodes.

# Routing design concepts (Distributed)

Update two vectors:

$$D_i = \begin{bmatrix} d_{i1} \\ \cdot \\ \cdot \\ d_{iN} \end{bmatrix} \qquad S_i = \begin{bmatrix} S_{i1} \\ \cdot \\ \cdot \\ S_{iN} \end{bmatrix}$$

D*i = Delay vector of node I*
*dij = current estimate of minimum delay from node i to node j*
N *= no of nodes in the network*
*Si = successor node vector for node i*
*sij = the next node in the current Minimum delay route for i to j*



# Conti …

$$d_{kj} = \min_{i \in A} \left[ d_{ij} + l_{ki} \right]$$

$$s_{kj} = i$$

Using i that minimizes the preceding Expression

Where
A = set of neighbor nodes for k
$l_{ki}$ = current estimates of delay from k to i

| Destination | Delay ($D_1$) | Next node ($S_1$) |
|---|---|---|
| 1 | 0 | — |
| 2 | 2 | 2 |
| 3 | 5 | 3 |
| 4 | 1 | 4 |
| 5 | 6 | 3 |
| 6 | 8 | 3 |

(a) Node 1's Routing table before update

| $D_2$ | $D_3$ | $D_4$ |
|---|---|---|
| 3 | 7 | 5 |
| 0 | 4 | 2 |
| 3 | 0 | 2 |
| 2 | 2 | 0 |
| 3 | 1 | 1 |
| 5 | 3 | 3 |

(b) Delay vectors sent to node 1 from neighbor nodes

| Destination | Delay | Next node |
|---|---|---|
| 1 | 0 | — |
| 2 | 2 | 2 |
| 3 | 3 | 4 |
| 4 | 1 | 4 |
| 5 | 2 | 4 |
| 6 | 4 | 4 |

$l_{1,2} = 2$
$l_{1,3} = 5$
$l_{1,4} = 1$

(c) Node 1's routing table after update and link costs used in update

**Figure 12.6 Original ARPANET Routing Algorithm**

Example: k=1, j=5
A=2,3,4
i=3 => 5(l1,3) + 1(d3,5) = 6
i=4 => 1 (l1,4) + 1 (d4,5) = 2
i=2 => 2 (l1,2) + 3 (d 2,5) = 5
Dkj (1,5) = 2 Skj = 4

# First generation

- **Estimated link delay ($l_{ki}$) is just a queue length for the link**
  - Outgoing link favored links with shorter queues
  - Queues change very quickly
  - Distributed perception of the shortest route could change while packet is in en route
    - Packet looking for low congestion rather than aiming the destination
    - Algorithm did not consider the speed of the link but queue length only, so preference was not given to the fast links

# 2nd generation

- **Rather than queue length, using delay as performance criterion**
  - Delay is measured directly
  - Arrival time and departure time of packet is time stamped at the node
    - > Total delay (for successful acknowledgement) = (departure time-arrival time) + transmission time and propagation delay
    - > Every 10 sec node computes average delay on outgoing links
    - > If significant difference in delay use flooding to transmit to others
    - > Each node maintains an estimate of delay on every network link

# 3rd generation

- 2nd generation more responsive and stable than 1st generation
- It was assumed that **packet delay is good predictor of the link delay**
  - Good strategy, if good correlation between reported values and those experience after re-routing
    - > High correlation for light to moderate traffic
    - > No correlation for high traffic

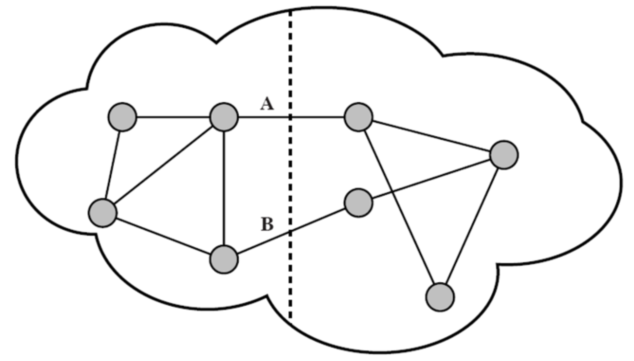



Figure 12.8 Packet-Switching Network Subject to Oscillations

# Conti …

- **Oscillation between links as a result**
  - This oscillation wastes capacity under heavy load
  - This could lead to congestion in the network
  - More routing tables updates – more control traffic at heavy load conditions
  - Every node was trying to obtain best route
    - > Solution: Give average route a good path rather than all routes the best path
      - Only change the way link costs are calculated

# How to achieve this

- **With single-server queuing model: measured delay is transformed to estimate of link utilization:**

$$\rho = \frac{2(T_s - T)}{T_s - 2T}$$

p = link utilization,
T= measured time
Ts = service time (network-wide average packet size (600 bits)/data rate of the link),

- Average utilization from previous estimates of utilization:

$$U(n+1) = 0.5\rho(n+1) + 0.5U(n)$$

# Conti …

- The link cost is taken as function of average utilization, which gives reasonable estimate of cost while avoiding oscillation
  - Estimated utilization = > delay = > cost

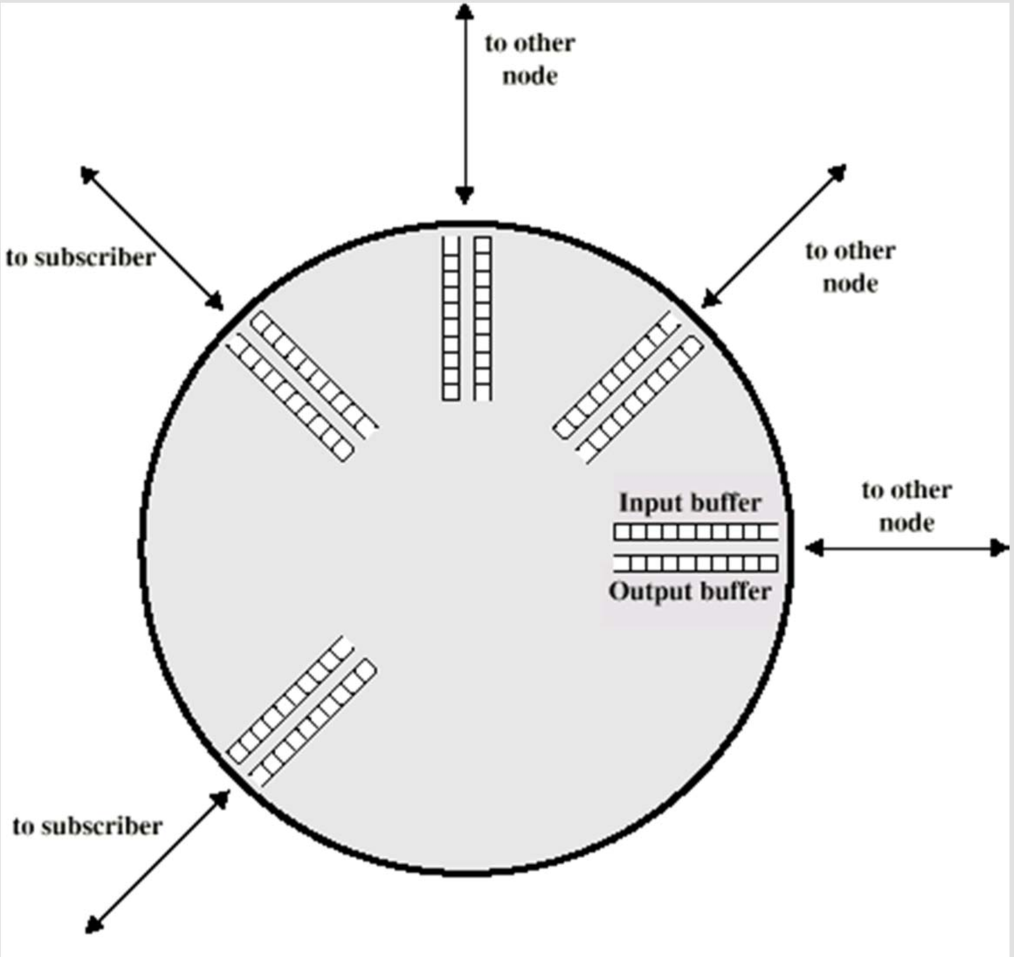
# Network Congestion

- Network congestion is a situation where the network or part of the network is overloaded with packets, the network performance is degraded dramatically.
- Ex: Cinema exit through a small door after a movie show. The result is no movement or very slow movement.
- Causes for the network congestion.
  - Intermediate node (switching node) is too slow to process and forward packets, transmission line has plenty of capacity
  - The switching node fast, but the capacity of the transmission line is limited

# Conti …

- A packet-switching network is a network of queues
- Queues build up and eventually no space for new arriving packets
- Outgoing channel is too slow, the packets in the outgoing queue build up. Eventually the outgoing queue would be full and put pressure on the incoming queue since the packets in the incoming queue cannot be forwarded

# Queues at a Node



# Effects of Congestion

- Packets arriving are stored at input buffers
- Routing decision made
- Packet moves to output buffer
- Packets queued for output transmitted as fast as possible
  - Statistical time division multiplexing
- If packets arrive to fast to be routed, or to be output, buffers will fill
- Can discard packets
- Can use flow control
  - Can propagate congestion through network

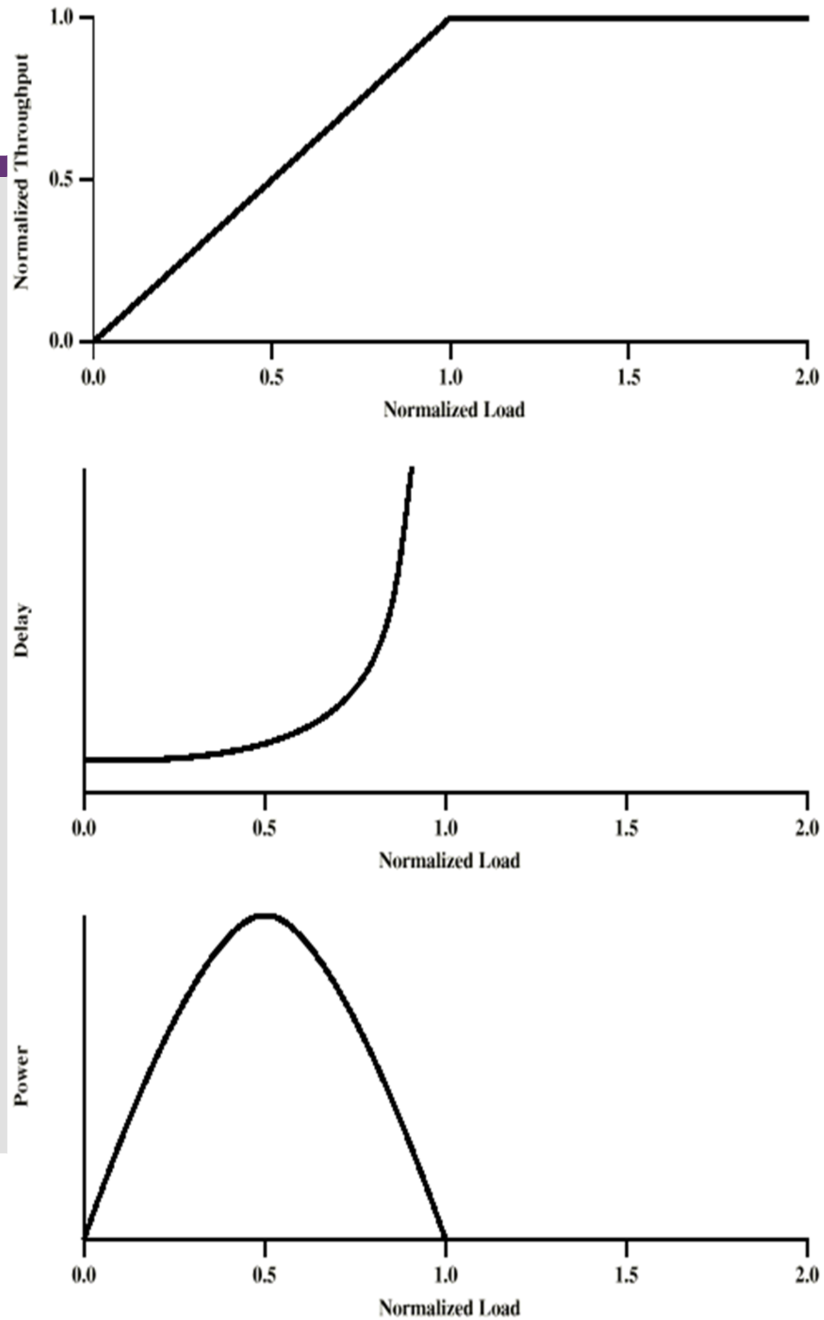
# Conti …

- Congestion control aims to keep number of packets below level at which performance falls off dramatically
- Data network is a network of queues
- Generally 80% utilization is critical

# Ideal Performance

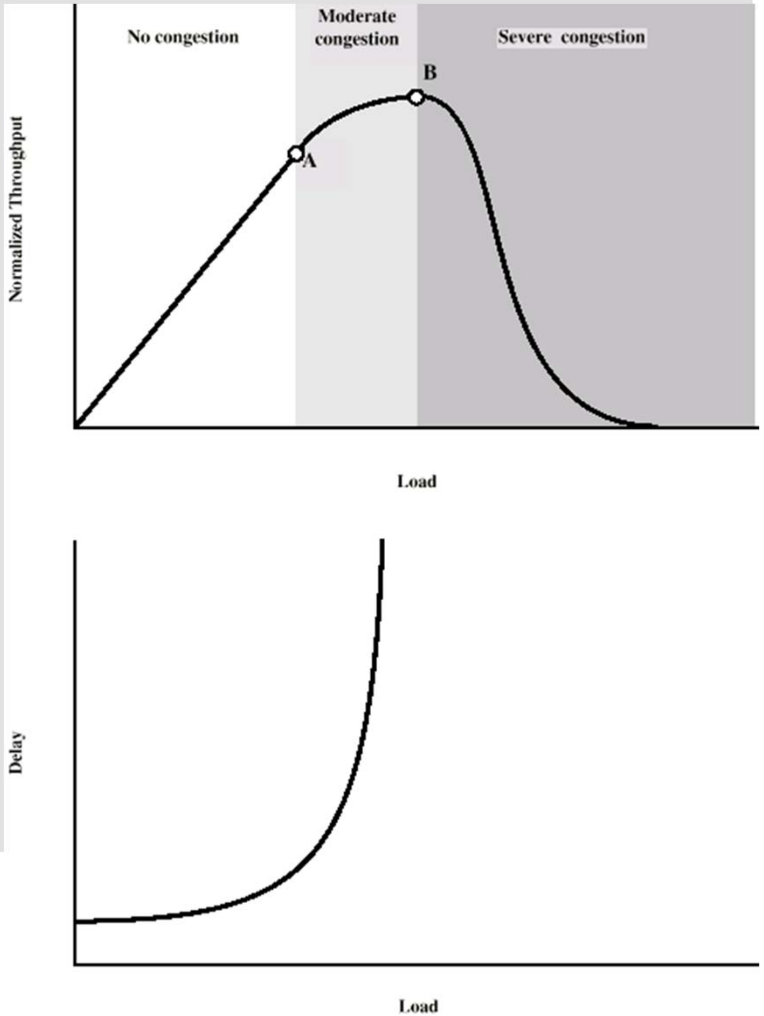
- Normalized load
- Power = Throughput/delay
- Normalized throughput
  - **No of packet delivered**
- Normalized load
  - **No of packets transmitted**

- Normalized to throughput



# Effects of Congestion - No Control

- Limited size of buffers




MONASH University
Information Technology

# Congestion and Flow Control

- **Congestion control:**
  - Makes sure that subnet is able to carry the required load
  - It is global issue
- **Flow control:**
  - Relates to point to point traffic between a given sender and receiver
  - Direct contact between sender and receiver for coordination

# Congestion Control Techniques

- **Open-loop**
  - Solution to problem by good design without regard to the current state of the network.
  - It decides when to accept new traffic
  - When to discard packets
  - Makes scheduling decisions at various points in the network.

# Conti …

- **Closed-loop**
  - Based on the concept of feedback loop
    - Monitors the system to detect when and where congestion occurs
    - Measured in terms of percentage of packets discarded, the average queue lengths, the number of retransmission, the average packet delay.
  - Pass this information to nodes, such as the packet sources, where action can be taken.
  - Adjust system operation to correct the problem.

# Congestion Prevention

- **Backpressure**
  - Source should reduce load by halting the transmission
- **Choke Packet**
  - Packet sent to source node to reduce the transmission (ICMP)
- **Implicit Congestion Signaling**
  - Detection of delayed packet delivery
  - Or detection of packet discarding
- **Explicit Congestion Signaling**
  - Network informs end systems about growing congestion
  - Steps are taken

# Conti …

- **Policies**
  - Implemented at design level to prevent congestion
    - > Retransmission of packets – data link layer level
    - > Flow control policy with acknowledgement
    - > Choice between virtual circuit and datagram affects congestion – Network layer
    - > Routing algorithm – spreading traffic to other lines
    - > Retransmission, Out of order policy - transport layer
    - > Timeout determination (long/short or adjustable)

# Traffic Management

- **Congestion control is concerned efficient use of network in heavy load**
- **Packet discarding should take into account issues:**
  - Fairness
    - > Last packet in - first discard may be not good
    - > Could maintain queues for each logical connection or source-destination
    - > Queues with highest traffic will suffer high packet discard
  - Quality of service
    - > May want different treatment for different connections
    - > Multimedia no delay but inaccuracy accepted, file transfer needs accuracy but time delay accepted
  - Reservations
    - > e.g. ATM
    - > Traffic contract between user and network

# Congestion Control in Packet Switched Networks

- **Send control packet to some or all source nodes**
  - Requires additional traffic during congestion
- **Rely on routing information**
  - Link utilization information to other nodes to make routing decisions, this information could be used to influence how fast the packets are produced
- **End to end probe packets (time stamped) to measure delay between two end points**
  - Adds to overhead
- **Add congestion info to packets as they cross nodes**
  - Either backwards or forwards

# Frame Relay
# Congestion Control

- Minimize discards
- Maintain agreed QoS
- Minimize probability of one end user monoply
- Simple to implement
  - Little overhead on network or user
- Create minimal additional traffic
- Distribute resources fairly
- Limit spread of congestion
- Operate effectively regardless of traffic flow
- Minimum impact on other systems
- Minimize variance in QoS:
  - individual logical connections should not experience sudden degradation due to congestion

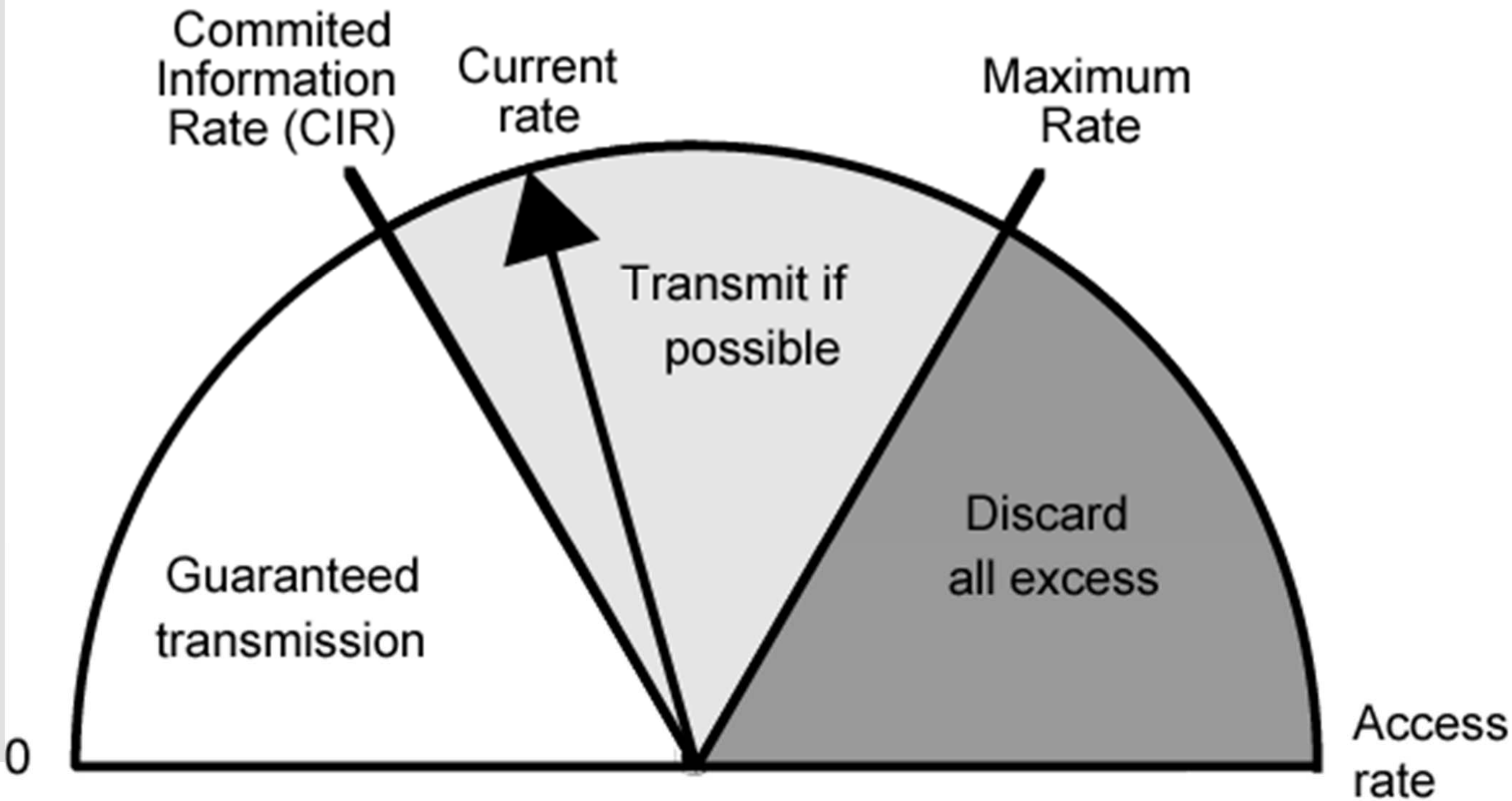
# Techniques

- **Discard strategy:**
  - discard frames, fairly to all users
- **Congestion avoidance using:**
  - Explicit signaling
- **Congestion recovery**
  - To prevent network from collapse
  - Message conveyed by protocols e.g LAPF or TCP to Implicit signaling mechanism

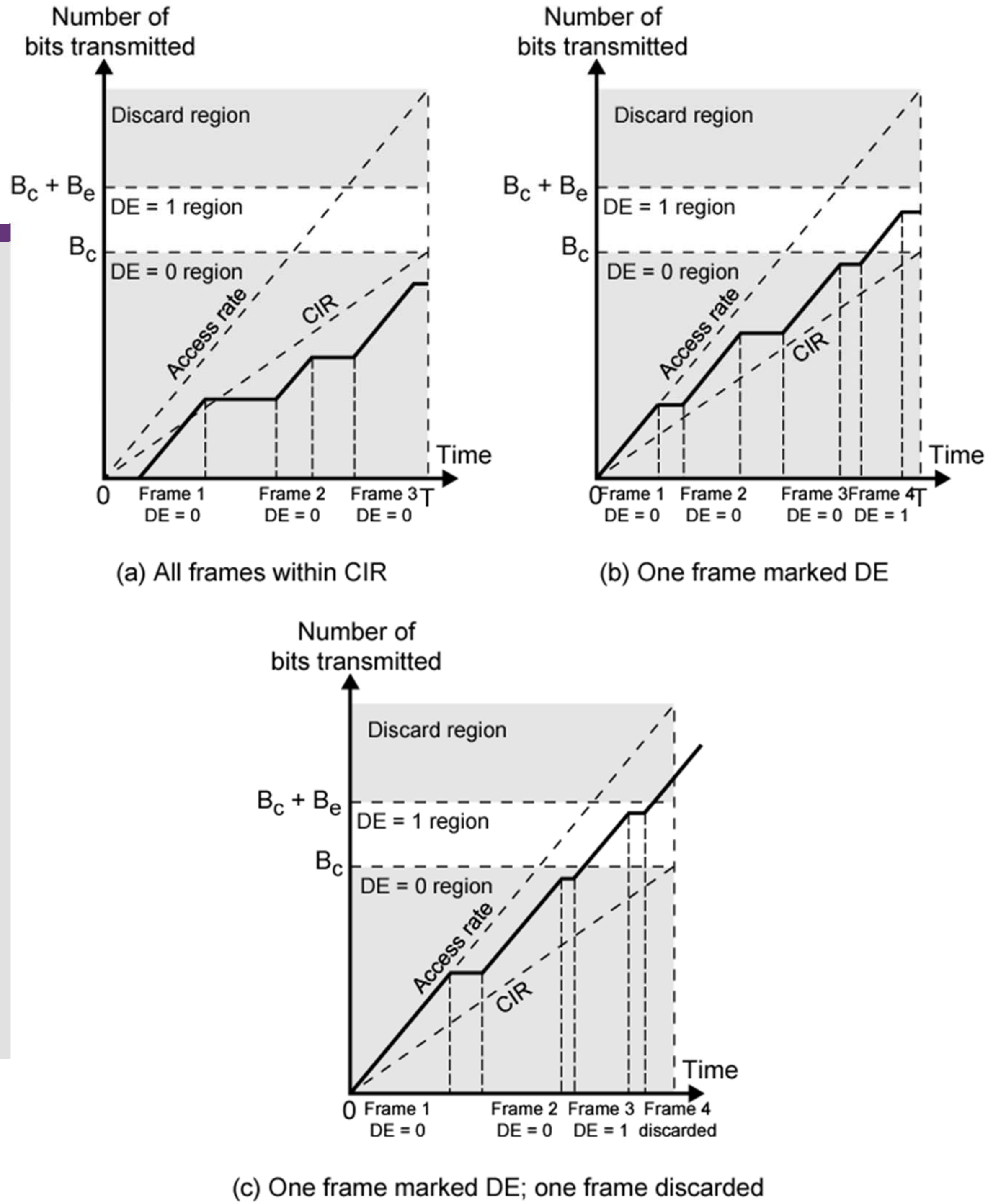
# Traffic Rate Management

- **Must discard frames to cope with congestion**
  - Arbitrarily, no regard for source
  - No reward for restraint so end systems transmit as fast as possible making the problem worse
  - Committed information rate (CIR) concept for fair allocation of resources in Frame relay
    - > Agreed to support this traffic (CIR) but Not guaranteed
    - > Data in excess of this liable to discard
    - > Aggregate CIR should not exceed physical data rate
- **CIR provides a way which frame to discard in congestion situation**
- **Discrimination is indicated by bit discard eligibility (DE) in the LAFP frame**
- **Metering function (at user station) determines, if data rate < CIR then do not change DE bit, so do not discard**
  - If data rate > CIR then set DE in the frame, frame may be discarded

# Operation of CIR



# Relationship Among Congestion Parameters




MONASH University
Information Technology

# Explicit Signaling

- Network alerts end systems of growing congestion
- Backward explicit congestion notification
- Forward explicit congestion notification
- Frame handler monitors its queues
- May notify some or all logical connections
- User response
  - Reduce rate

# Summary

- **Network layer and its functions**
- **Routing**
- **Network congestion and prevention**
- **Routing and congestion play important role in Internetworking; this will be completed in next lecture**

- **Next Lecture**
  - Internetworking