Transport Layer Contd

Internet Technologies
COMP90007

Reading Update

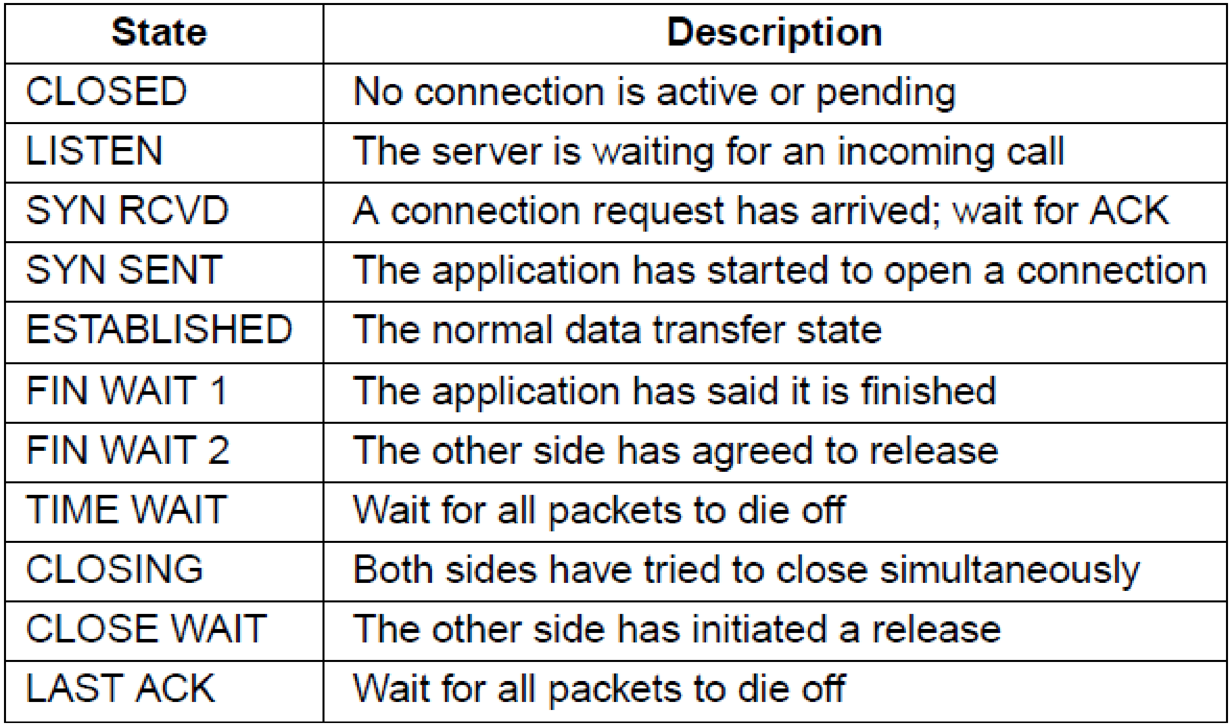
- Some of the content for this week, i.e., on congestion control, requires reading further sections that you may have skipped from the previous Network Layer Chapter

TCP Details Contd.
Connection and Release Implementation

- Connections established using three-way handshake
- Connections released with symmetric release
- Timers used for lost connection releases

TCP Connection Management: Full Set of States

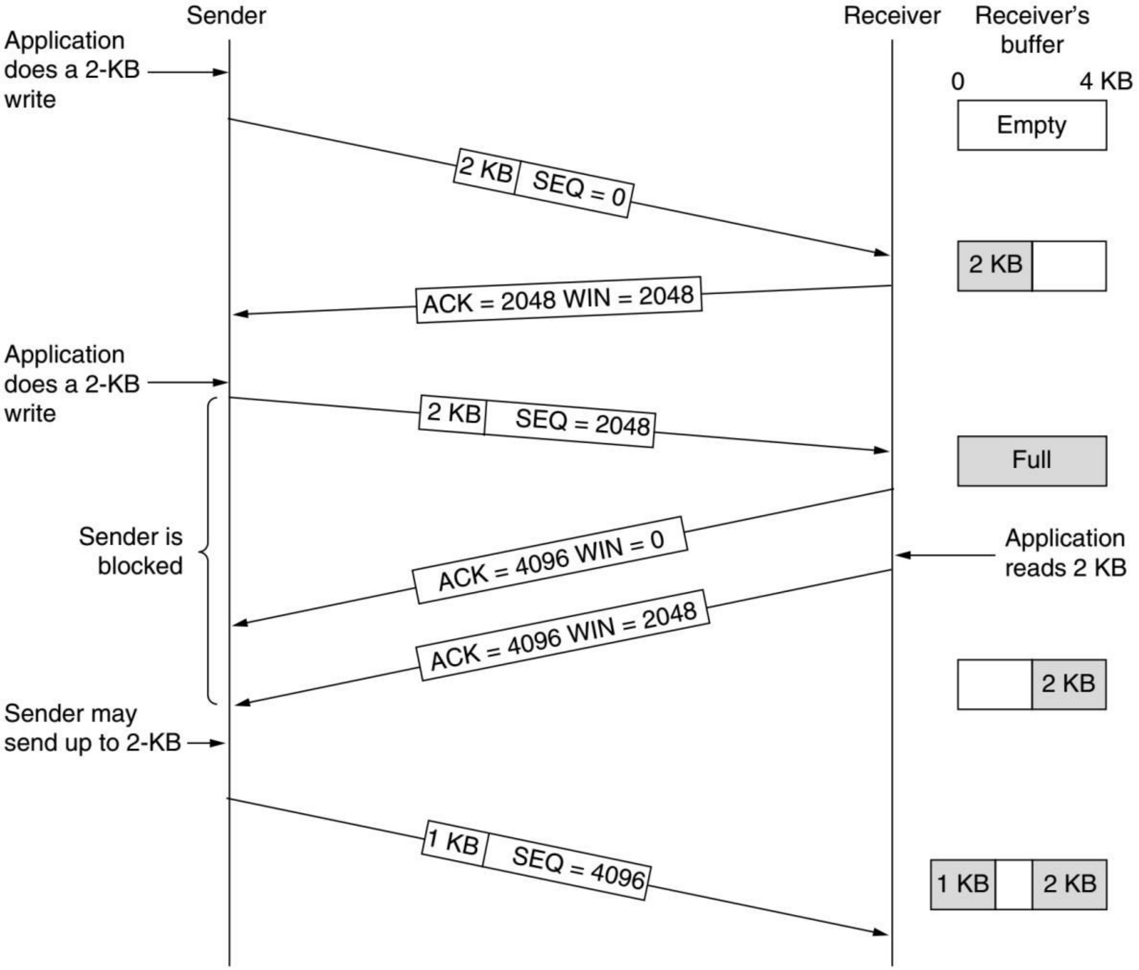
- The full TCP connection finite state machine has more states than the simple example we saw earlier but essence is the same
- The following states are worth noting from real world implementations



State	Description
CLOSED	No connection is active or pending
LISTEN	The server is waiting for an incoming call
SYN RCVD	A connection request has arrived; wait for ACK
SYN SENT	The application has started to open a connection
ESTABLISHED	The normal data transfer state
FIN WAIT 1	The application has said it is finished
FIN WAIT 2	The other side has agreed to release
TIME WAIT	Wait for all packets to die off
CLOSING	Both sides have tried to close simultaneously
CLOSE WAIT	The other side has initiated a release
LAST ACK	Wait for all packets to die off

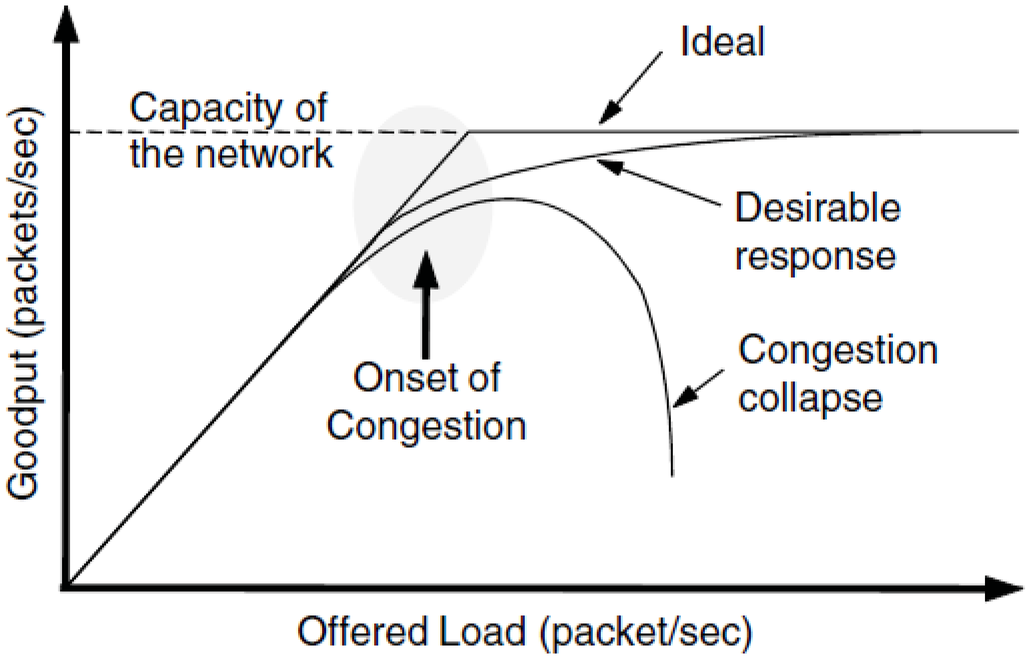
TCP Transmission Example

- TCP acknowledges bytes
- <u>*Receiver advertises window*</u> based on available buffer space



Next: How to Deal with Congestion

- Congestion results when too much traffic is offered; performance does not stabilize but rather degrades due to loss/retransmissions – THIS IS REALLY BAD!



Congestion Control vs Flow Control

- Flow control is an issue for point to point traffic, primarily concerned with preventing sender transmitting data faster than receiver can receive it
- Congestion control is an issue affecting the ability of the subnet to actually carry the available traffic, thus more in a global context

Load Shedding

- When congestion control mechanisms fail, load shedding is the key remaining possibility - **drop packets**
- **Else network performance collapses**
- In order to ameliorate impact, applications can mark certain **packets as priority** to avoid discard policy
- This is done at routers at Network Layer

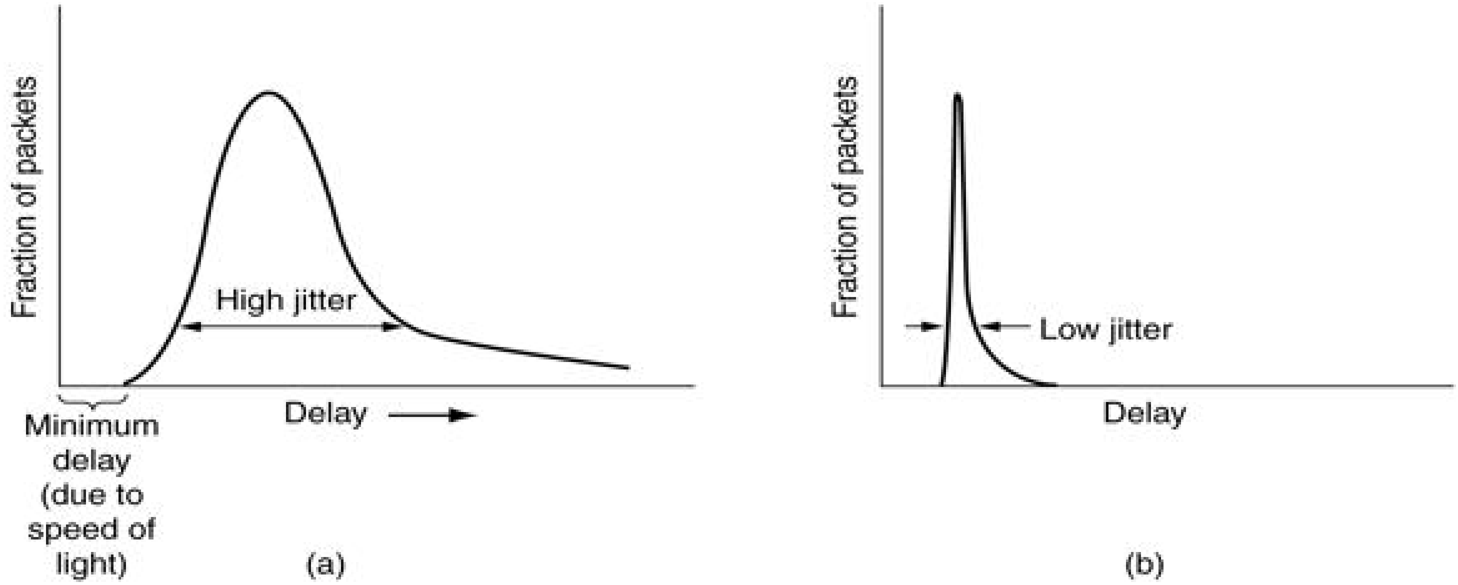
What happens when network is not delivering properly

- Quality of Service becomes low
- **Expected network performance** is an important criterion for a wide range of network applications
- Some **engineering techniques are available to guarantee QoS** (Quality of Service)
- 4 dimensions to watch out for:

  bandwidth, reliability, delay, jitter

Lets Look at Jitter Before Going Further

- Jitter is the **variation in packet arrival times**
  - a) high jitter
  - b) low jitter



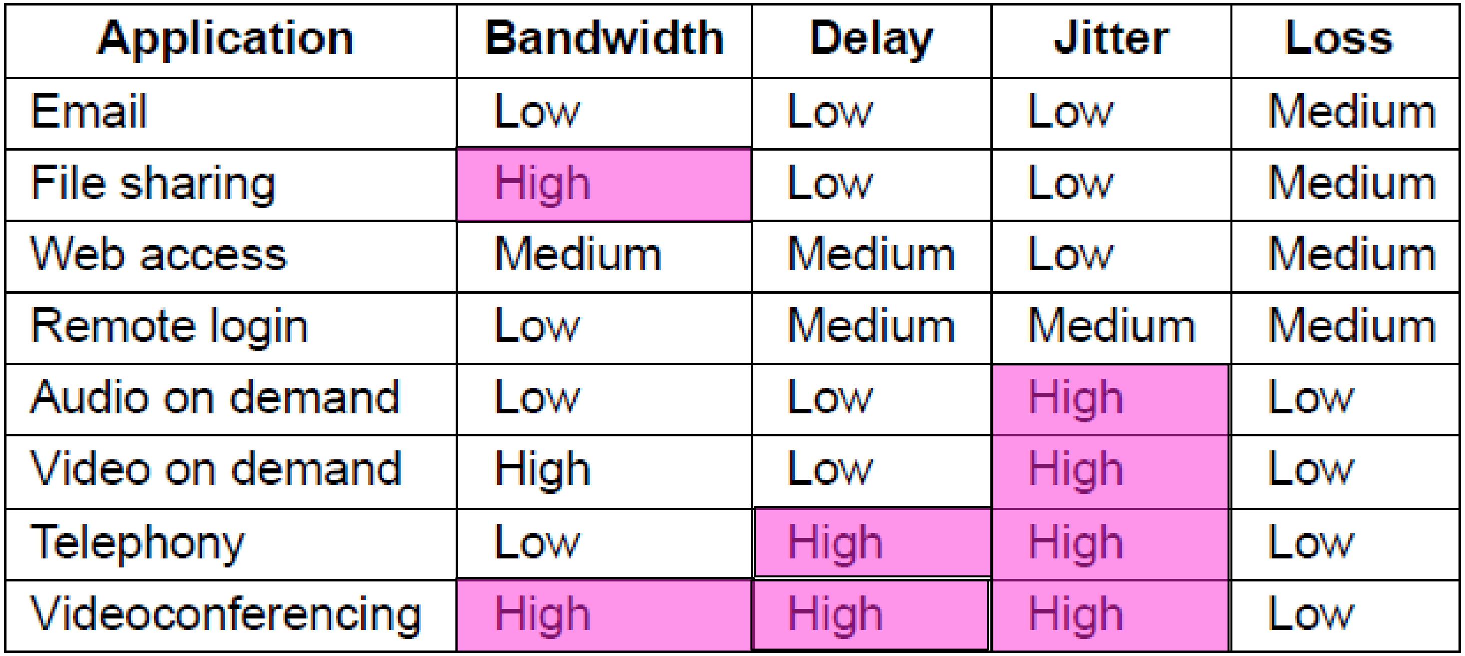
Mechanisms for Jitter Control

- Jitter is an issue for some applications
- Jitter can be contained by **determining the expected transit time** of a packet
- Packets can be **shuffled at each hop in order to minimise jitter** - slower packets sent first, faster packets wait in a queue
- For certain applications jitter control is extremely important as it directly affects the **quality perceived by the application user**

QoS Requirements

- Different applications care about different properties
  - We want all applications to get what they need

“High” means a demanding the requirement!

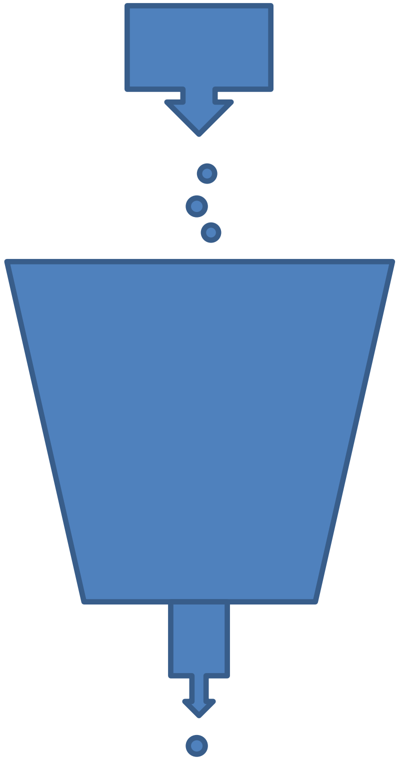


Application	Bandwidth	Delay	Jitter	Loss
Email	Low	Low	Low	Medium
File sharing	High	Low	Low	Medium
Web access	Medium	Medium	Low	Medium
Remote login	Low	Medium	Medium	Medium
Audio on demand	Low	Low	High	Low
Video on demand	High	Low	High	Low
Telephony	Low	High	High	Low
Videoconferencing	High	High	High	Low

Techniques Worth Mentioning for QoS

- **Over-provisioning**
  - more than adequate buffer, router CPU, and bandwidth (expensive and not scalable)
- **Buffering**
  - buffer received flows before delivery - increases delay, but smoothes out jitter, no effect in reliability or bandwidth
- **Traffic Shaping**
  - regulate the average rate of transmission and burstiness of transmission
  - ***leaky bucket : a cornerstone method***
  - **token bucket**

Leaky Bucket



Large **bursts** of traffic is buffered and smoothed while sending at network layer on sending host

Techniques Related to QoS Contd

- **Resource reservation**
  - reserve bandwidth, buffer space, CPU in advance
- **Admission control**
  - routers can decide based on traffic patterns whether to accept new flows, or reject/reroute them
- **Proportional routing**
  - traffic for same destination split across multiple routes
- **Packet scheduling**
  - create queue(s) based on priority etc
  - fair queuing, weighted fair queueing