Transport Layer Contd

COMP90007

Internet Technologies

1

What is the key problem if network is not delivering properly:

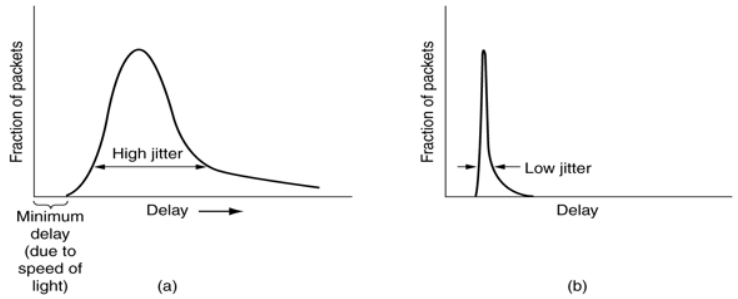
- Quality of Service becomes low
- **Expected network performance** is an important criterion for a wide range of network applications
- Some **engineering techniques are available to guarantee QoS** (Quality of Service)
- 4 things to watch out for:

  bandwidth, reliability, delay, jitter

2

Jitter is Interesting/New

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



3

Mechanisms for Jitter Control

- Jitter is important for some application??
- 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 mainly directly affects the **quality perceived by the application user**

4

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				
File sharing				
Web access				
Remote login				
Audio on demand				
Video on demand				
Telephony				
Videoconferencing				

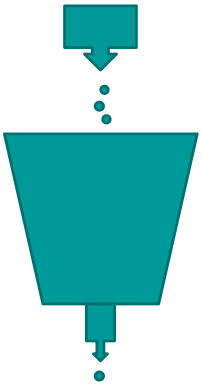
5

Techniques for Achieving QoS #1

- **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**
  - **token bucket**

6

Leaky Bucket



Large **bursts** of traffic is buffered and smoothed while sending

E.g. can be done at host sending data

7

Techniques for Good QoS #2

- **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

8

TCP and Congestion Control

- When networks are overloaded, congestion occurs, potentially affecting all layers
- Although lower layers (data and network) attempt to ameliorate congestion, in reality **TCP impacts congestion most significantly** because TCP offers best methods to reduce the data rate, and hence reduce congestion itself

9

Congestion Control: Design

- Two different problems exist
  - network capacity and receiver capacity
  - these should be dealt with separately, but compatibly
- The sender maintains two windows actually
  - Window described by the receiver
  - Congestion window
- Each regulates the number of bytes the sender can transmit – the maximum transmission rate is the **minimum of the two windows**

10

TCP and Congestion Control Contd

- TCP adopts a defensive stance:
  - At connection establishment, a **suitable window size is chosen by the receiver based on its buffer** size
  - If the sender is constrained to this size, then **congestion problems will not occur due to buffer overflow** at the receiver itself, but may still **occur due to congestion within the network**

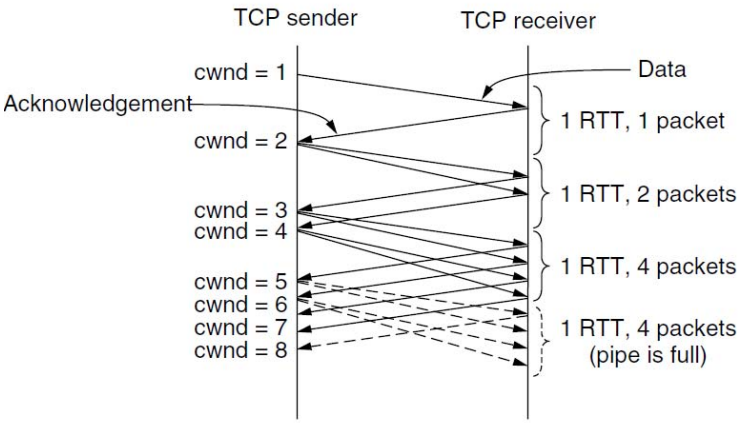
11

Incremental Congestion Control: Slow Start

- On connection establishment, the **sender initializes the congestion window to a size**, and transmits one segment
- If this segment is acknowledged before the timer expires, **the sender adds another segment's worth of bytes to the congestion window**, and transmits two segments
- As **each new segment is acknowledged**, the congestion window is increased by **one more segment**
- In effect, each set of acknowledgements doubles the congestion window - which **grows until either a timeout occurs or the receiver's specified window is reached**

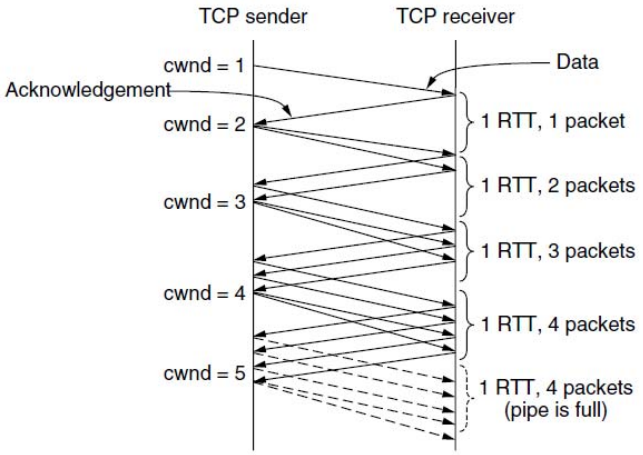
12

Slow Start



13

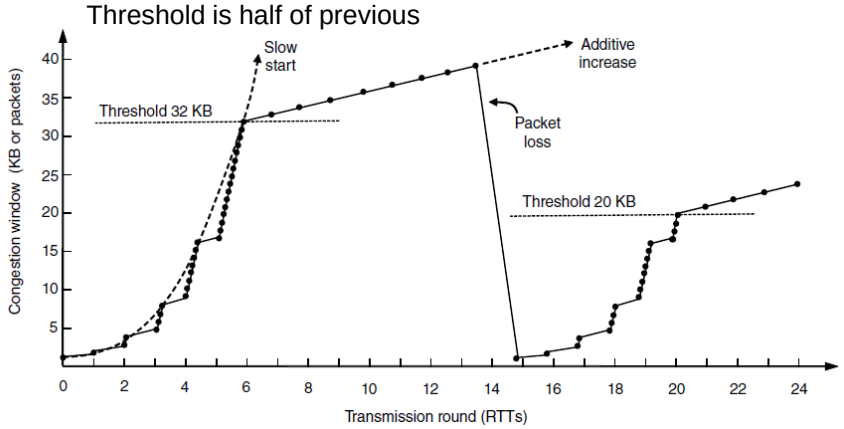
Additive increase



14

Internet Congestion Control Illustrated

Slow start followed by additive increase (TCP Tahoe)
Threshold is half of previous



15