National University of Computer & Emerging Sciences

CS 3001 – COMPUTER NETWORKS

Lecture 15

Chapter 3

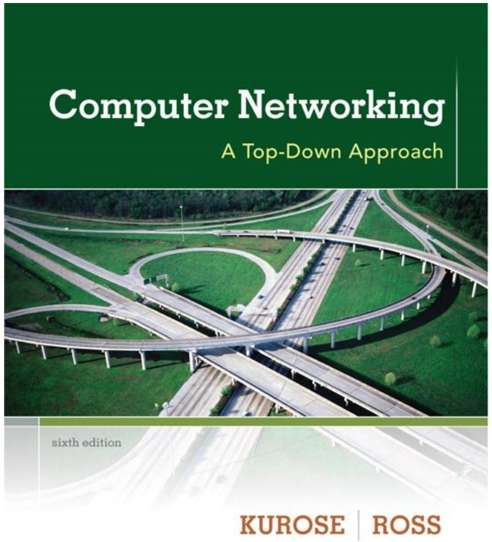
13th October, 2022

Nauman Moazzam Hayat

nauman.moazzam@lhr.nu.edu.pk

Office Hours: 02:30 pm till 06:00 pm (Every Tuesday & Thursday)

Chapter 3
Transport Layer



A note on the use of these ppt slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- If you use these slides (e.g., in a class) that you mention their source (after all, we'd like people to use our book!)
- If you post any slides on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

Computer Networking: A Top Down Approach
6th edition
Jim Kurose, Keith Ross
Addison-Wesley
March 2012

Chapter 3 outline

Principles of congestion control

congestion:

- informally: "too many sources sending too much data too fast for *network* to handle"
- different from flow control!
- manifestations:
  - long delays (queueing in router buffers)
  - lost packets (buffer overflow at routers)

- a top-10 problem!

Approaches towards congestion control

two broad approaches towards congestion control:

end-end congestion control:

- no explicit feedback from network
- congestion inferred from end-system observed loss, delay
- approach taken by TCP

network-assisted congestion control:

- routers provide feedback to end systems
  - single bit indicating congestion (SNA, DECbit, TCP/IP ECN, ATM)
  - explicit rate for sender to send at

Chapter 3 outline

3.1 transport-layer services

3.2 multiplexing and demultiplexing

3.3 connectionless transport: UDP

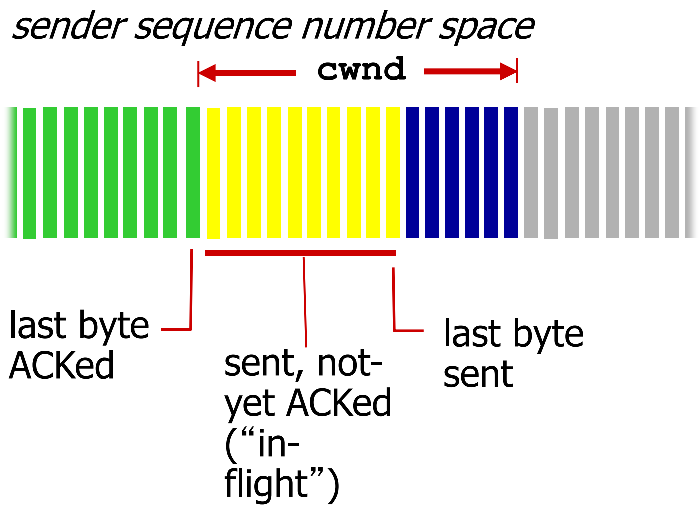
3.4 principles of reliable data transfer

3.5 connection-oriented transport: TCP
- segment structure
- reliable data transfer
- flow control
- connection management

3.6 principles of congestion control

3.7 TCP congestion control

TCP Congestion Control: details



- sender limits transmission:

(LastByteSent - LastByteAcked ≤ min{cwnd, rwnd} but ignore rwnd for this discussion, thus

`LastByteSent - LastByteAcked ≤ cwnd`

- **cwnd** is dynamic, function of perceived network congestion

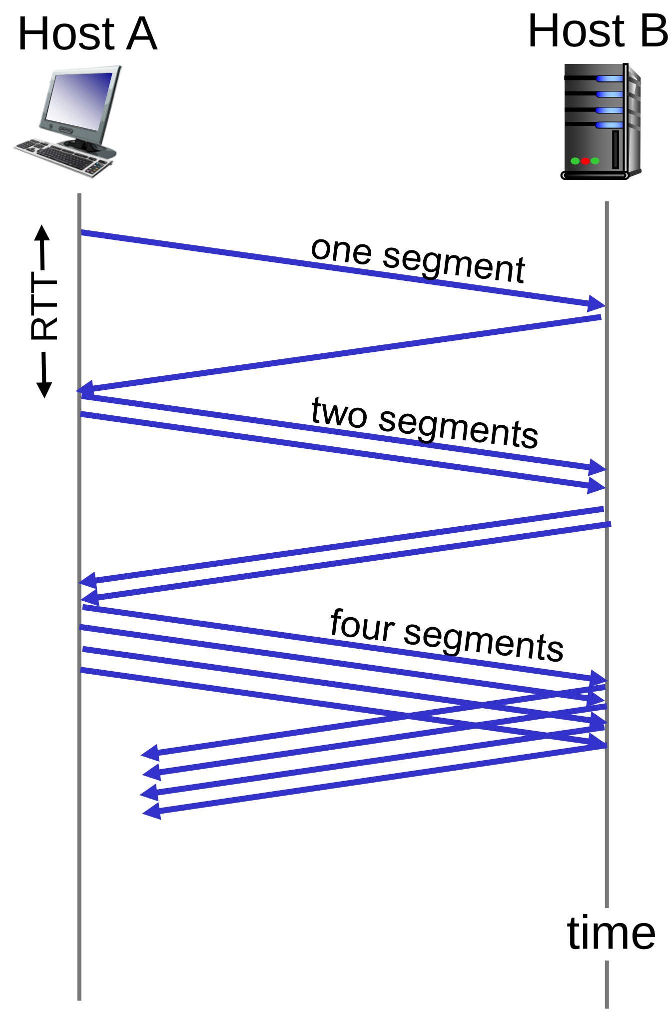
TCP sending rate:

- *roughly:* send cwnd bytes, wait RTT for ACKS, then send more bytes

$$\text{rate} \approx \frac{\text{cwnd}}{\text{RTT}} \text{ bytes/sec}$$

TCP Slow Start

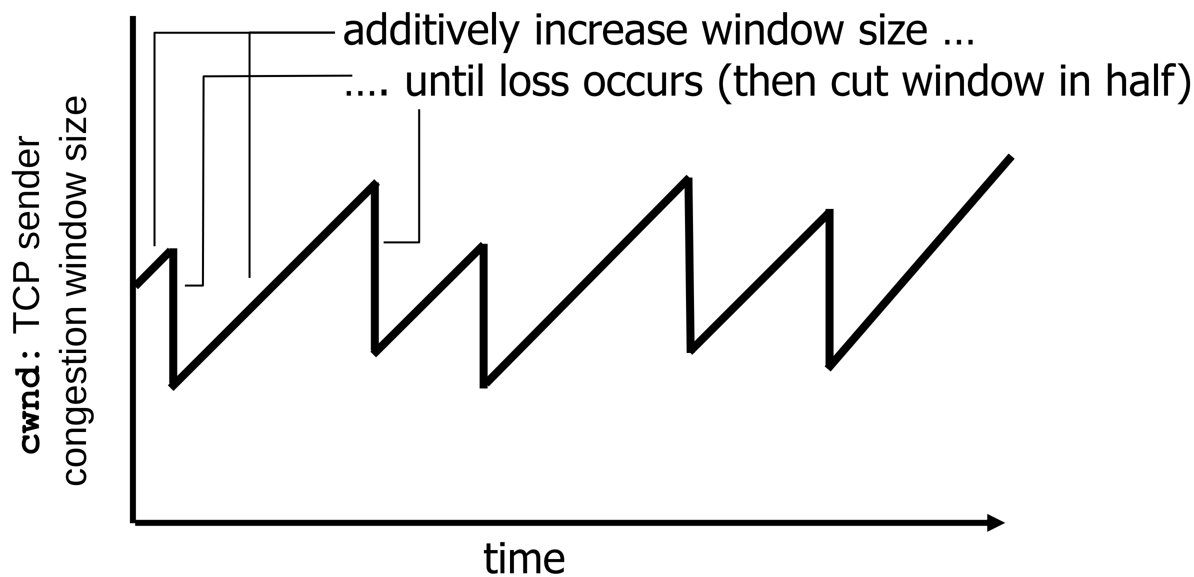
- when connection begins, increase rate exponentially until first loss event:
  - initially **`cwnd`** = 1 MSS
  - double **`cwnd`** every RTT
  - done by incrementing **`cwnd`** for every ACK received
- *summary:* initial rate is slow but ramps up exponentially fast



TCP congestion control: additive increase multiplicative decrease (AIMD, used in Congestion Avoidance)

- *approach:* sender increases transmission rate (window size), probing for usable bandwidth, until loss occurs
  - *additive increase:* increase **`cwnd`** by 1 MSS every RTT until loss detected
  - *multiplicative decrease:* cut **`cwnd`** in half after loss

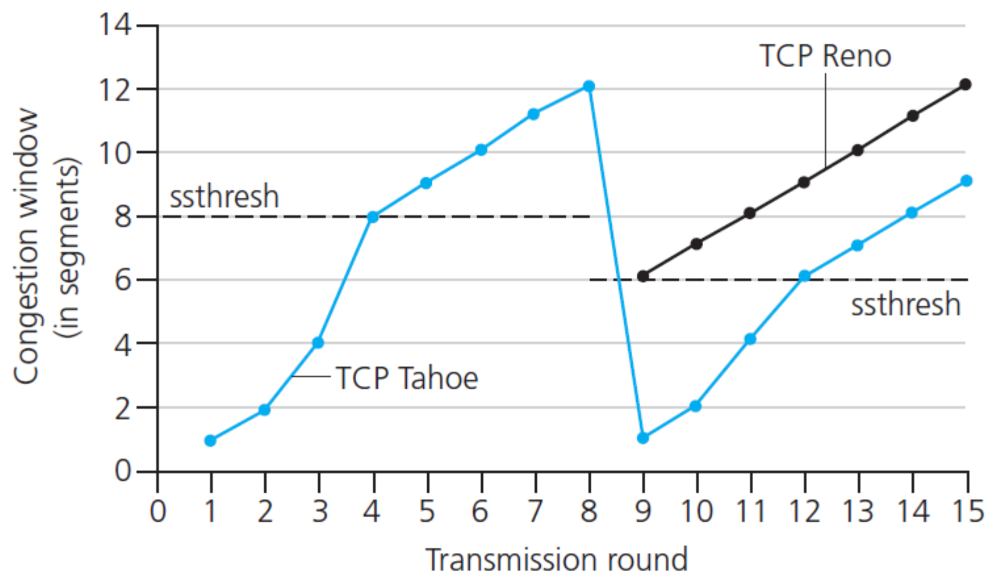
AIMD saw tooth behavior: probing for bandwidth



TCP: switching from slow start to CA

Q: when should the exponential increase switch to linear?

A: when **cwnd** gets to 1/2 of its value before timeout. (ssthresh)



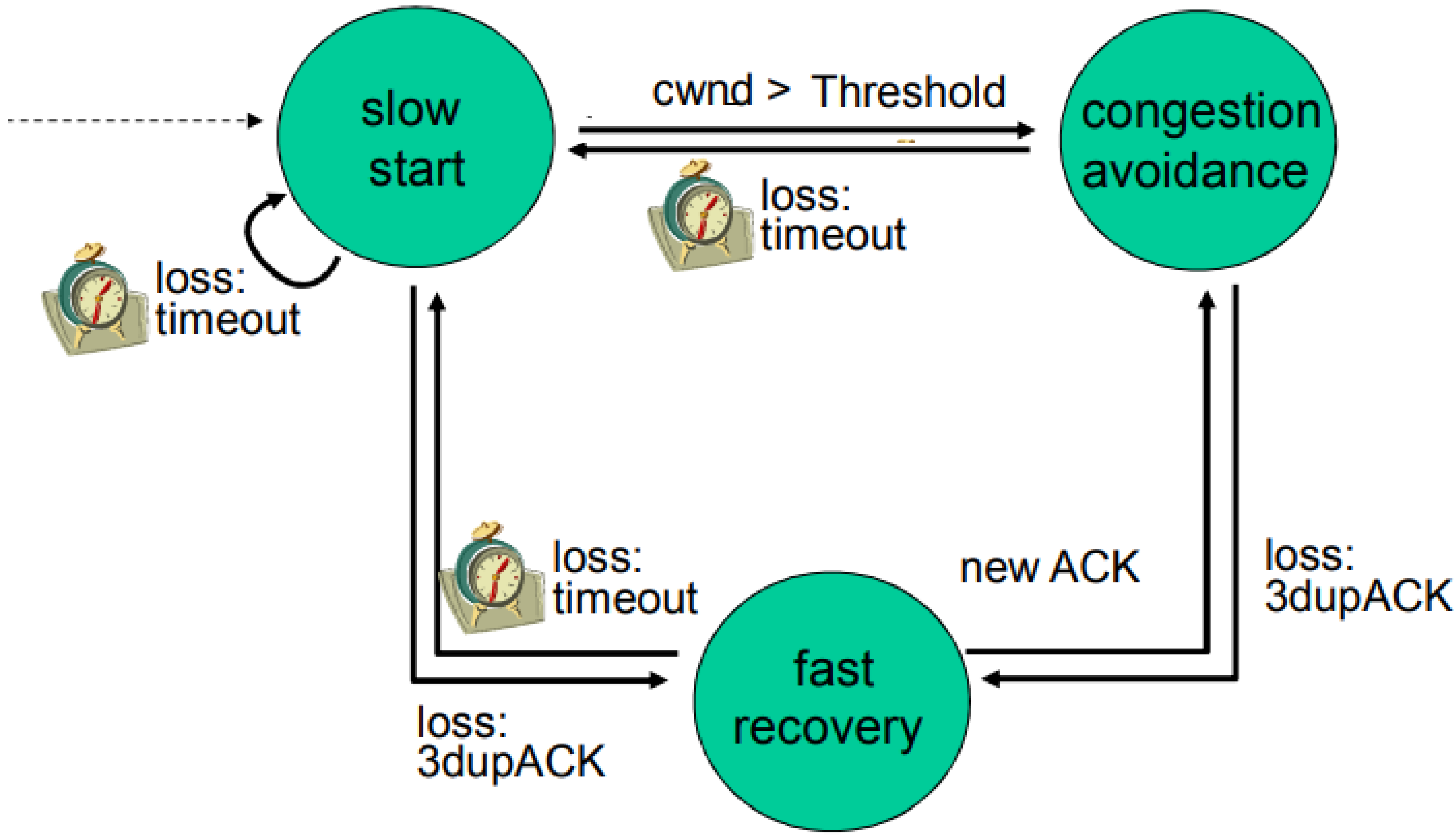
Implementation:

- variable **ssthresh**
- on loss event, **ssthresh** is set to 1/2 of **cwnd** just before loss event

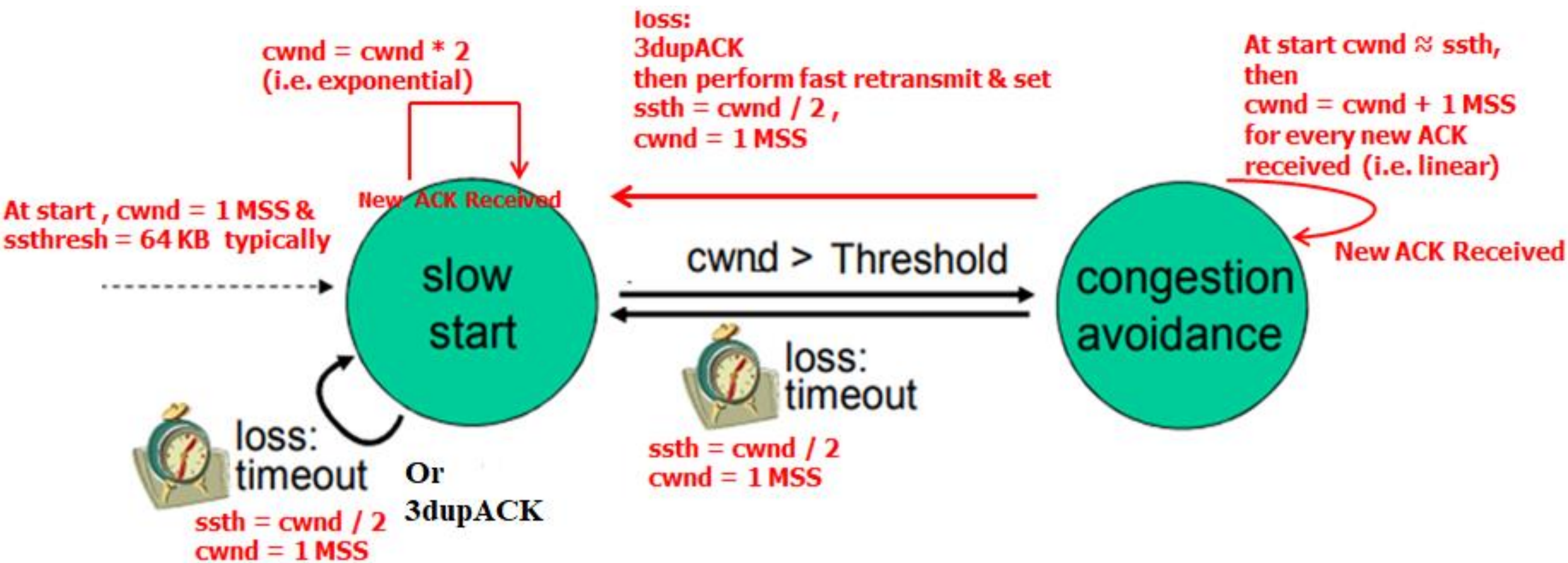
TCP: detecting, reacting to loss

- loss indicated by timeout: [both in TCP Tahoe (old)& TCP Reno (new)]
  - **`cwnd`** set to 1 MSS;
  - window then grows exponentially (as in slow start) to threshold, then grows linearly (AIMD used in Congestion Avoidance (CA))
- loss indicated by 3 duplicate ACKs: TCP RENO
  - dup ACKs indicate network capable of delivering some segments
  - **`cwnd`** is cut in half window then grows linearly
- TCP Tahoe always sets **`cwnd`** to 1 (timeout or 3 duplicate acks)

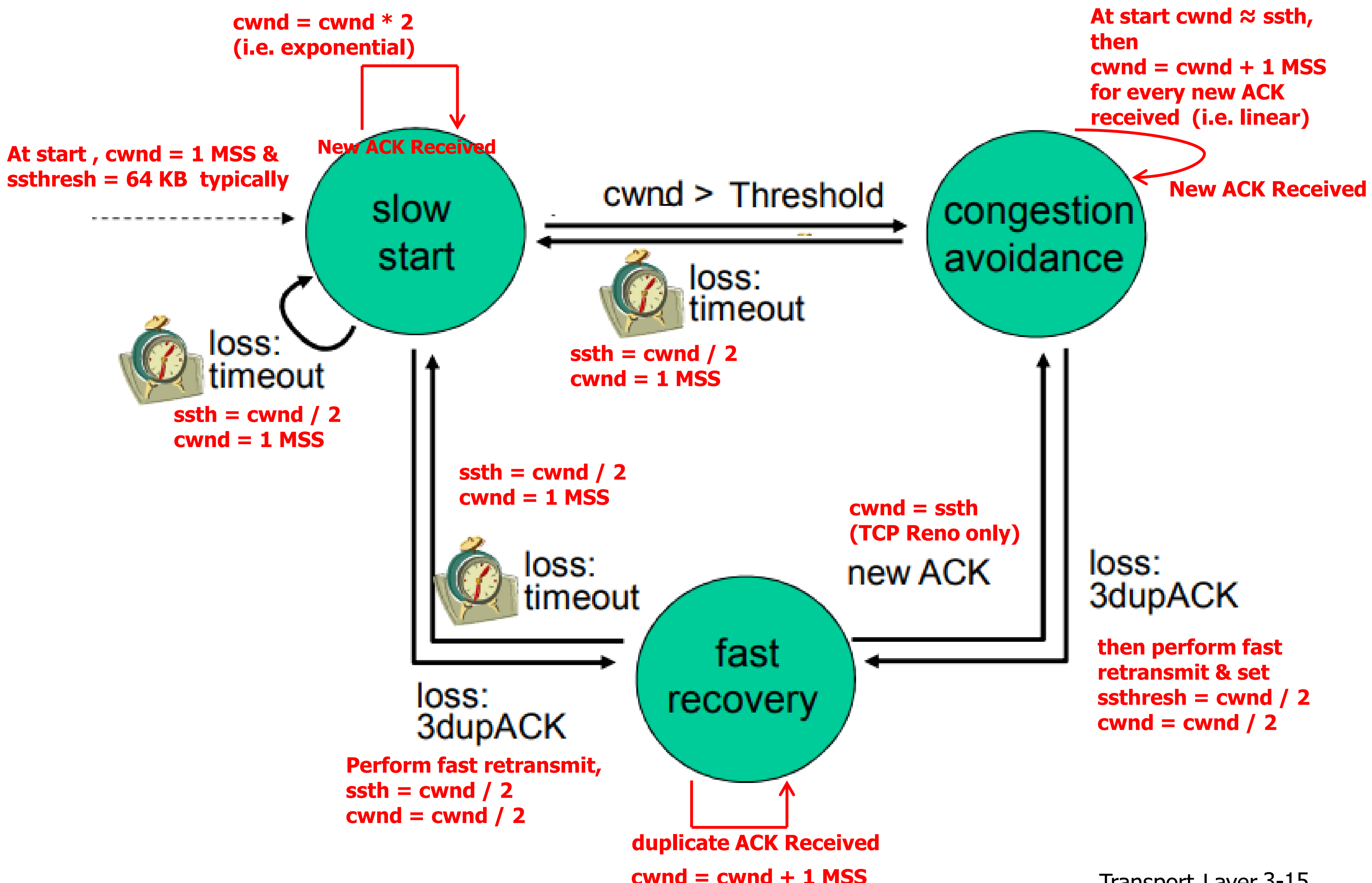
Summary: TCP Congestion Control



Summary: TCP Congestion Control (TCP Tahoe)



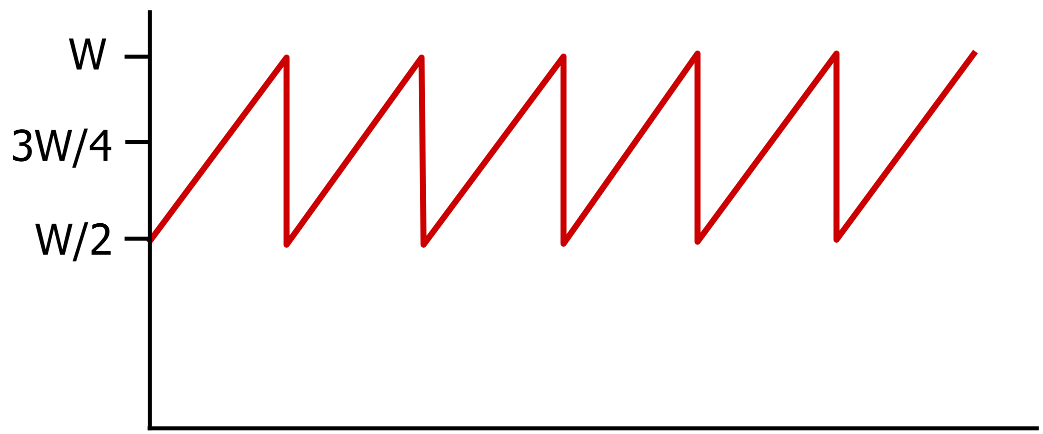
Summary: TCP Congestion Control (TCP Reno)



TCP throughput

- avg. TCP throughput as function of window size, RTT?
  - ignore slow start (as this phase is too short), assume always data to send
- W: congestion window (cwnd) size (measured in bytes) where loss occurs
  - avg. window size (# in-flight bytes) is ¾ W
  - avg. throughput is 3/4W per RTT

$$\text{avg TCP thruput} = \frac{3}{4}\frac{W}{RTT} \text{ bytes/sec}$$



Chapter 3: summary

- principles behind transport layer services:
  - multiplexing, demultiplexing
  - reliable data transfer
  - flow control
  - congestion control
- instantiation, implementation in the Internet
  - UDP
  - TCP

<u>next:</u>

- leaving the network "edge" (application, transport layers)
- into the network "core"

Assignement # 3 (Chapter – 3)

- *3rd Assignment will be uploaded on Google Classroom on Thursday, 13th October, 2022, in the Stream - Announcement Section (not the Classwork Section)*

- *Due Date: Tuesday, 18th October, 2022 (Handwritten solutions to be submitted during the lecture)*

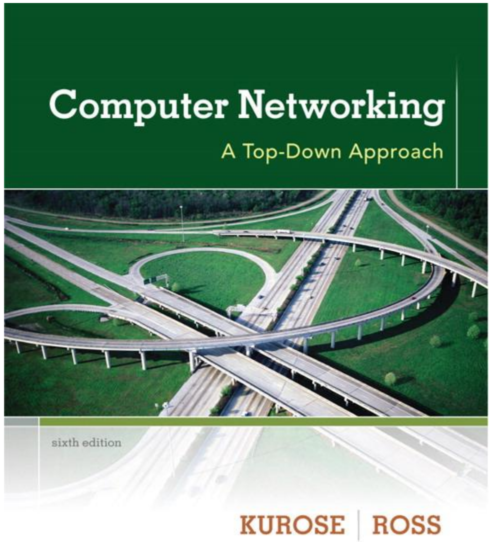
- *Please read all the instructions carefully in the uploaded Assignment document, follow & submit accordingly*

Quiz # 3 (Chapter – 3)

- *On: Thursday, 20th October, 2022 (During the lecture)*

- *Quiz to be taken during <u>own section class only</u>*

Chapter 4
Network Layer



A note on the use of these ppt slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- If you use these slides (e.g., in a class) that you mention their source (after all, we'd like people to use our book!)
- If you post any slides on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

Computer Networking: A Top Down Approach
6th edition
Jim Kurose, Keith Ross
Addison-Wesley
March 2012

Chapter 4: network layer

chapter goals:

- understand principles behind network layer services:
  - network layer service models
  - forwarding versus routing
  - how a router works
  - routing (path selection)
  - broadcast, multicast
- instantiation, implementation in the Internet

Chapter 4: outline

4.1 introduction

4.2 ~~virtual circuit and datagram networks~~

4.3 ~~what's inside a router~~

4.4 IP: Internet Protocol

- datagram format
- IPv4 addressing
- ICMP
- IPv6

4.5 routing algorithms

- link state
- distance vector
- hierarchical routing

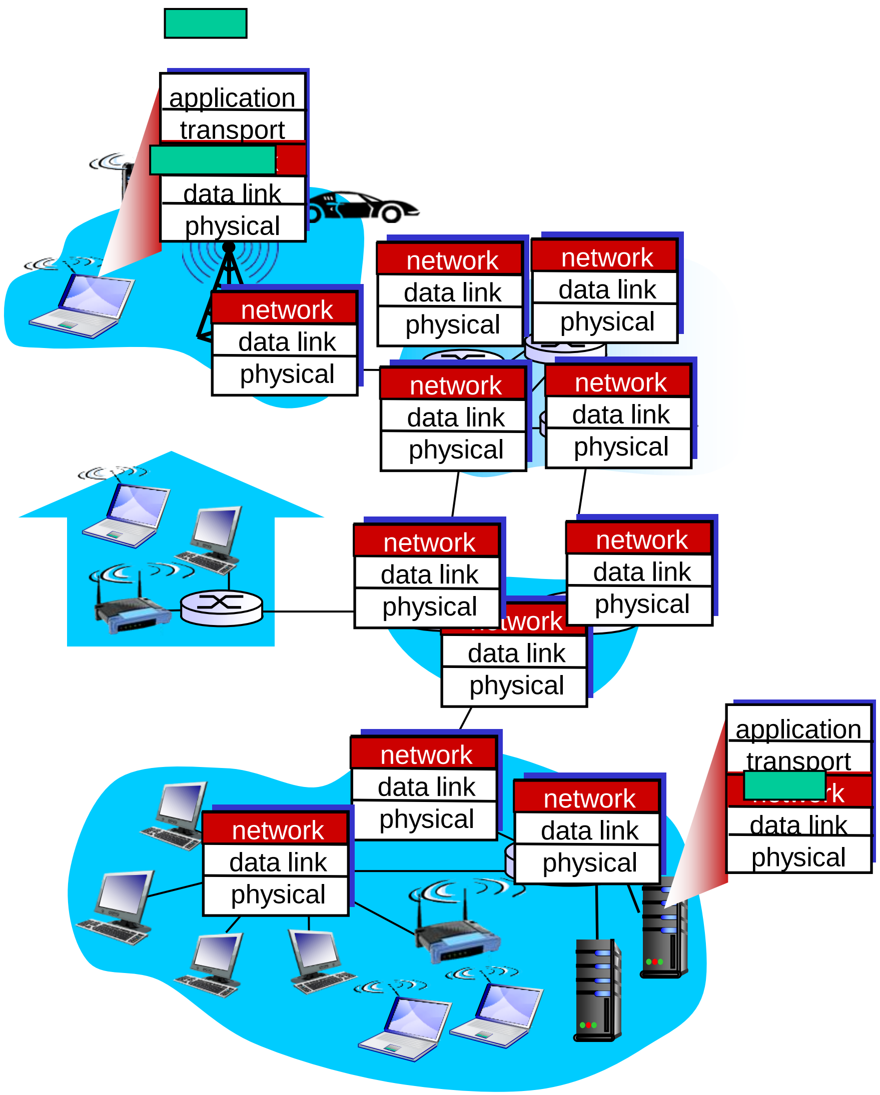
4.6 routing in the Internet

- RIP
- OSPF
- BGP

4.7 ~~broadcast and multicast routing~~

Network layer

- transport segment from sending to receiving host
- on sending side encapsulates segments into datagrams
- on receiving side, delivers segments to transport layer
- network layer protocols in *every* host, router
- router examines header fields in all IP datagrams passing through it
- No call setup at Network Layer
- No state about end-to-end connections



Two key network-layer functions

- *forwarding:* move packets from router’ s input to appropriate router output (transfer of a packet from an incoming link to an outgoing link within a single router)

- *routing:* determine route taken by packets from source to dest.
  - *routing algorithms*

analogy:

- *forwarding:* process of getting through single interchange

- *routing:* process of planning trip from source to dest

Interplay between routing and forwarding

4 billion IPv4 possible addresses, so instead of listing individual IP address, list range of addresses (aggregate table entries)

routing algorithm determines end-end-path through network

forwarding table determines local forwarding (to next hop) at this router

value in arriving packet's header