Wireless Communication Networks & Systems

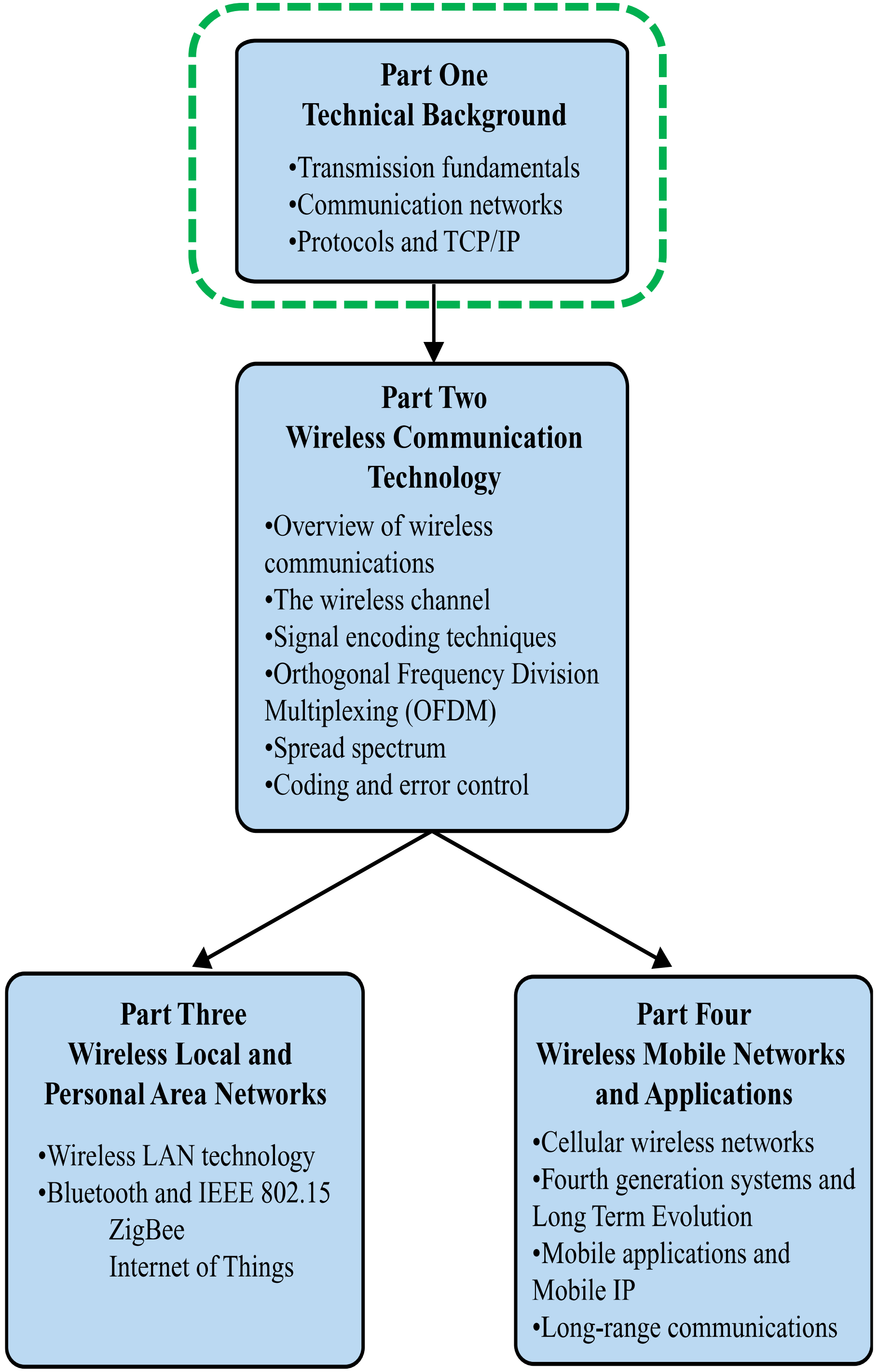
Spring 2025

Week No. 03

Course Learning Outcomes (CLOs):

CLO # 01	Demonstrate an in-depth understanding of wireless network system's architecture, protocols, and Services.	**Cog. 3**
CLO # 02	Explore advanced technologies and features in wireless networks related to coverage, capacity, interference management, and mobility.	**Cog. 3**
CLO # 03	Examine the evolution of Wi-Fi networks, highlighting architectural differences across its various standards.	**Cog. 4**
CLO # 04	Analyze key cellular concepts used in cellular networks and the architectural advancements in 5G and beyond.	**Cog. 4**

WCNS Module- I



Textbook Reference

<u>Module 01- Technical Background of Wireless Networks</u>

- Chapter 02 – The Signals and Transmission Fundamentals
- Chapter 03 – Communication Networks and QoS
- Chapter 04 – TCP/IP Network Model and & Admission/congestion Control

TCP/IP Network Model and & Admission/congestion Control

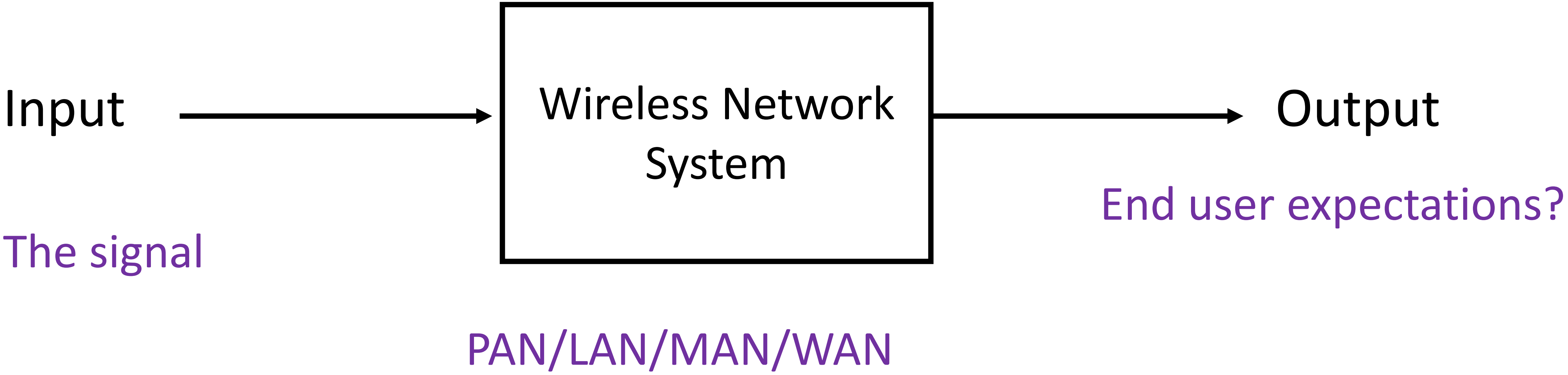
LEARNING OBJECTIVES

After studying this chapter, you should be able to:

- Describe the importance and objectives of a layered protocol architecture.
- Explain the process of protocol encapsulation where PDUs are handed to different layers and headers are added for control purposes.
- Describe the TCP/IP architecture and explain the functioning of each layer.
- Explain the need for internetworking.
- Describe the operation of a router within the context of TCP/IP to provide internetworking.

Wireless Networks Systems

- Input
- Output
- System’s Main building Blocks



Wireless Networks Systems - Output

- High Data rate - throughput
- Low delay
- High reliability, availability of good signals
- High reliability, low packet drop rate
- Support to provide a large number of users, Capacity
- Seamless Mobility

Generic Framework of QoS

Overprovisioning

Prioritization without guarantees

Prioritization with guarantees

Practical Implementation QoS Models

BE

InteServ.

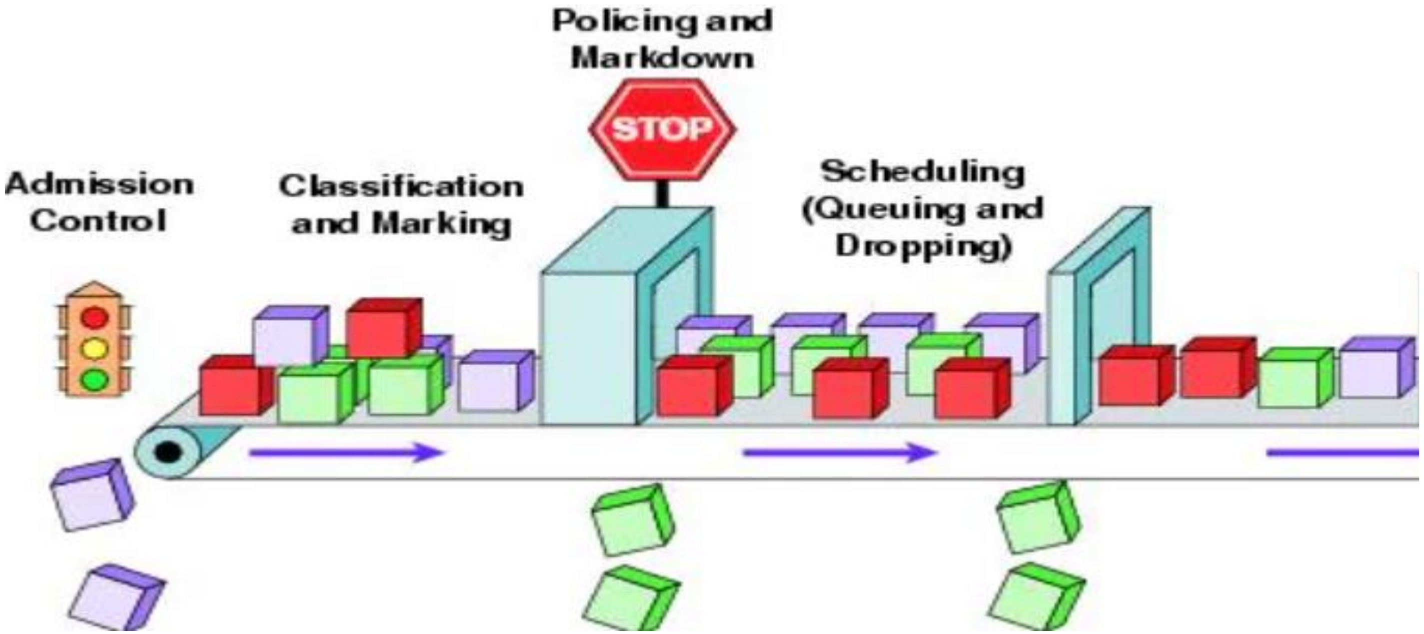
DiddServ.

Key techniques used in QoS Models

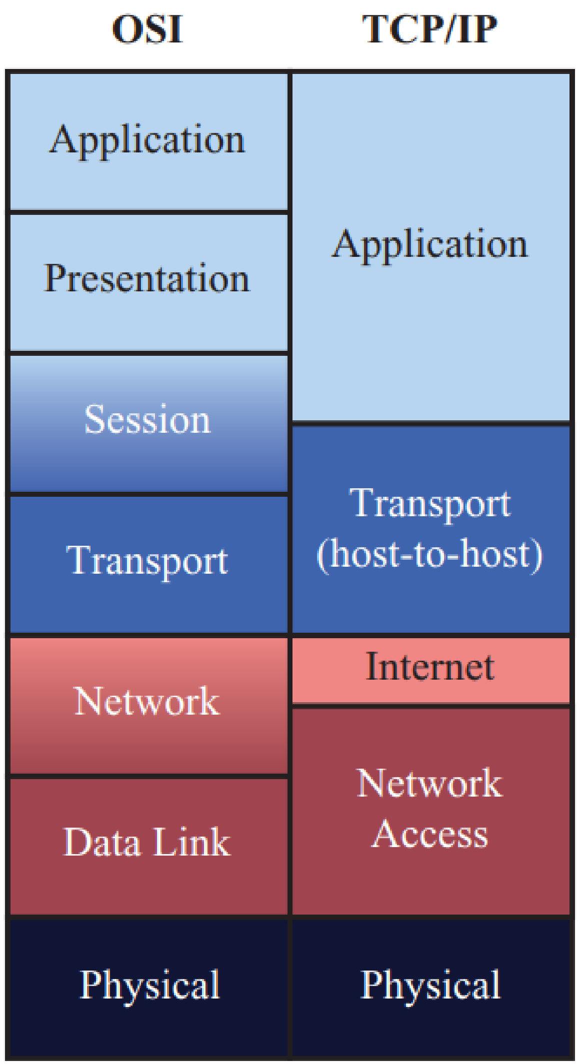
Queuing and Scheduling

Traffic Shaping or Policing

Admission Control



Layered Architecture TCI/IP and OSI Model



TCP/IP Model

- 4.1-The need for a protocol architecture
- 4.2- The TCP/IP protocol architecture
- 4.3- The OSI Model
- 4.4- The Interworking

Key Terms

application layer	internetworking	router
checksum	network layer	service access point (SAP)
frame check sequence (FCS)	physical layer	Transmission Control Protocol (TCP)
header	port	transport layer
internet	protocol	User Datagram Protocol (UDP)
Internet Protocol (IP)	protocol architecture protocol data unit (PDU)	
intranet IPv6		

TCP/IP Model

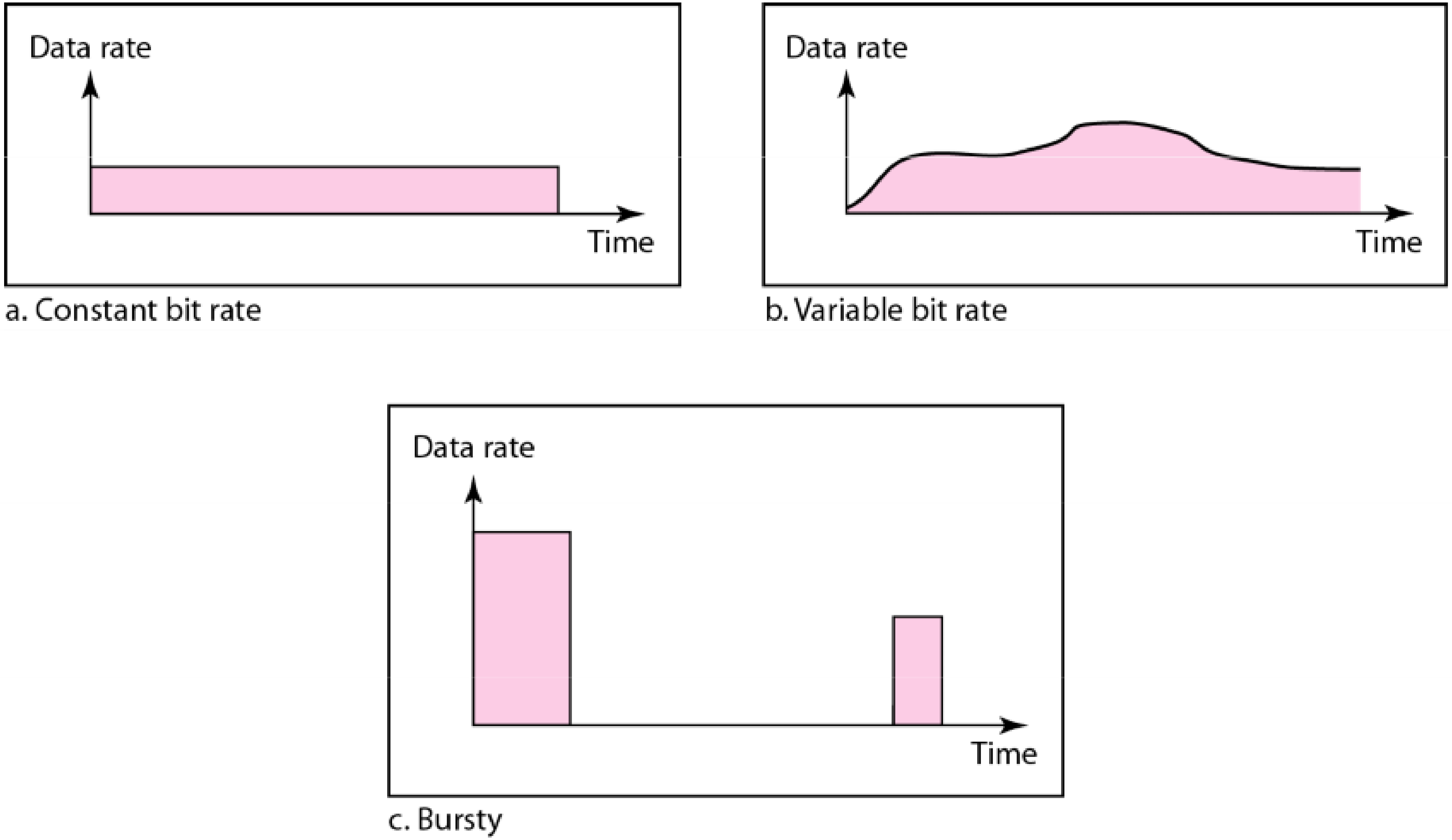
- Appendix 4A: The Internet Protocol
- Appendix 4B: The Transmission Control Protocol (TCP)
- Appendix 4C: The User Datagram Protocol (UDP)

Admission & Congestion Control
Transport Layer

The Transmission Control Protocol (TCP)

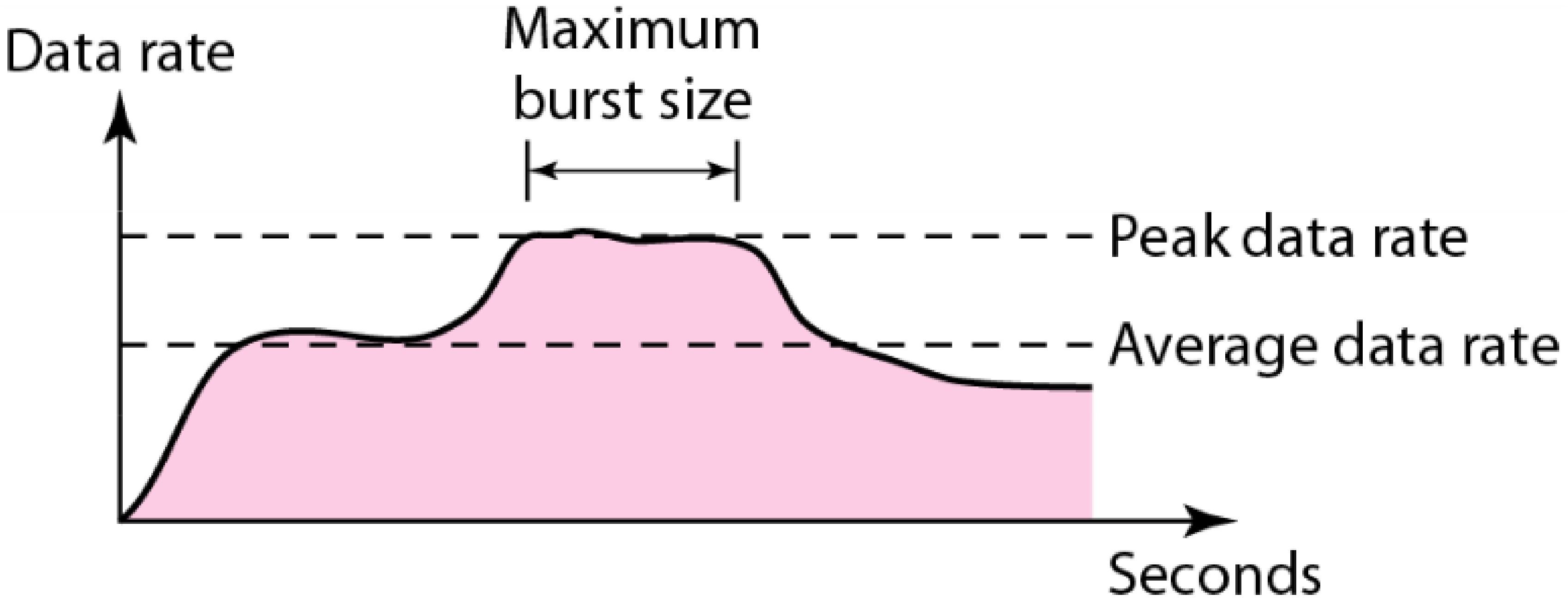
Data Traffic:

Traffic Profile:



Data Traffic:

Qualitative way to represent of data flow is called **Traffic descriptor**



Average Data Rate

Peak Data Rate

Maximum Burst size

Effective transmission Bandwidth: the bandwidth that the network needs to allocate to cater flow of data traffic. The effective bandwidth Is the function of the average data rate, and peak data rate, and maximum burst size.

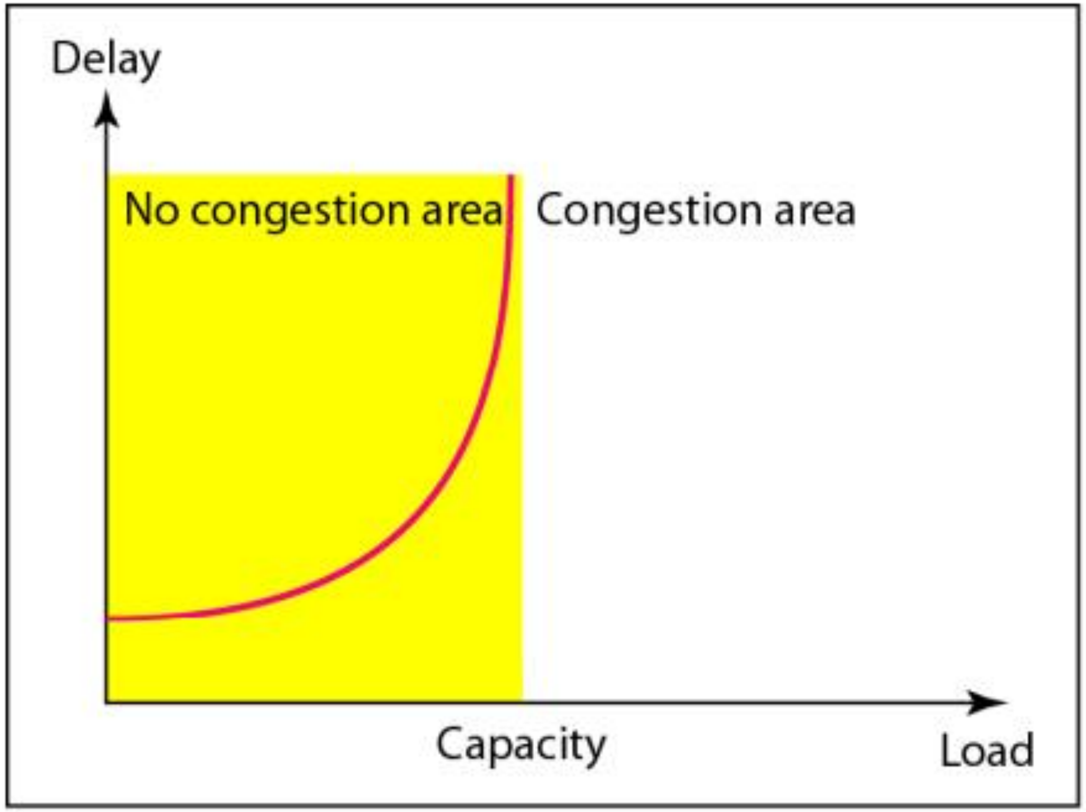
Network Congestion:

- A **network congestion** occurs if the load on the network, i.e., the number of packets sent to the network, is greater than its capacity. Here, the network's capacity means the number of packets the network can handle.

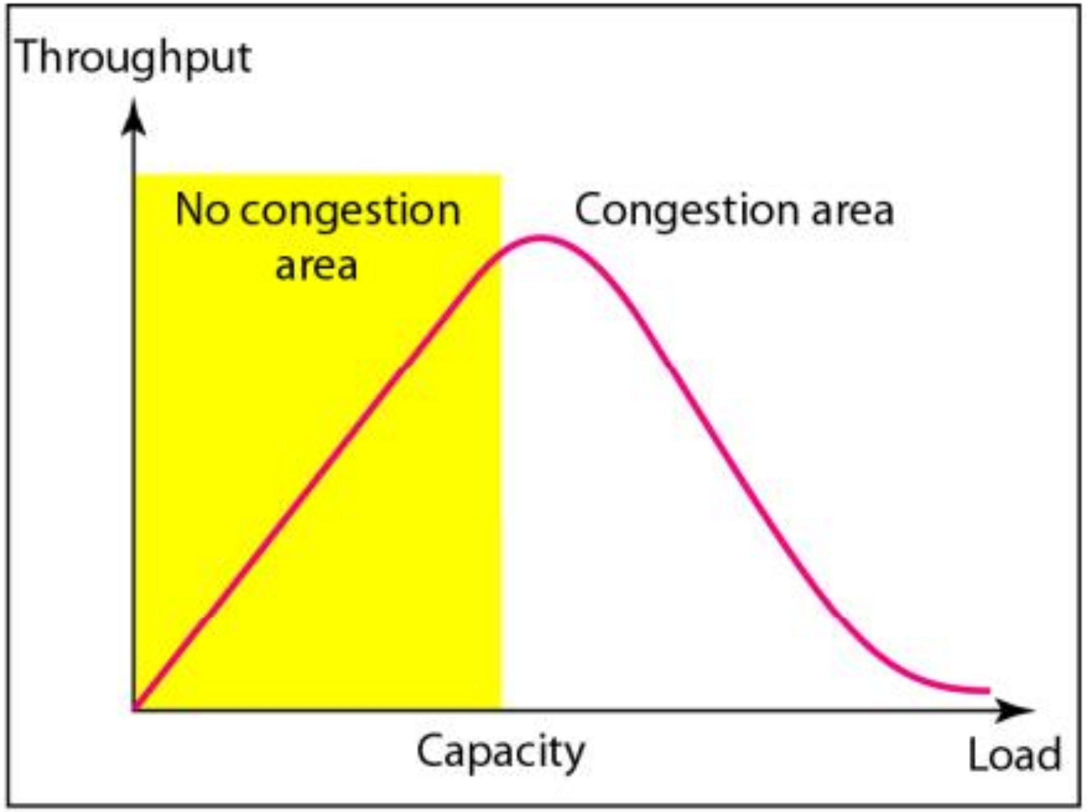
- **Congestion control** refers to the mechanism/technique for controlling congestion and keeping the load below to ensure QoS.

Congestion
long Queues
waiting time

Network Congestion:



a. Delay as a function of load



b. Throughput as a function of load

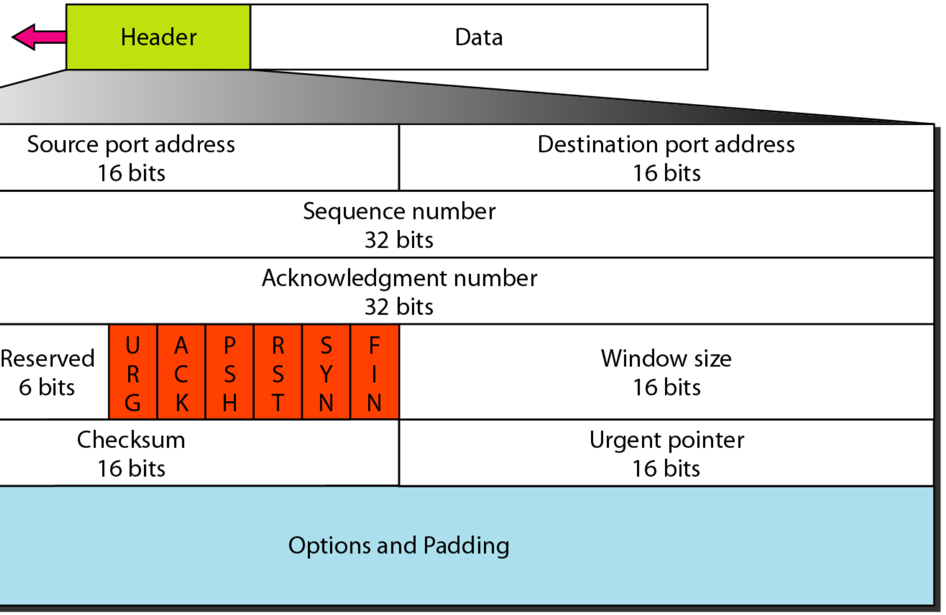
QoS: Packet Delay & Throughput

Data Flow & Congestion Control:

TCP window:

- In the flow control mechanism of TCP, the size of the send window is controlled by the receiver using the value of rwnd, which is advertised in each segment traveling in the opposite direction. The use of this strategy guarantees that it would never overflow the receiver.

- This, however, does not mean that the intermediate buffers, buffers in the routers, do not become congested. A router may receive data from more than one sender.



Data Flow & Congestion Control:

TCP window:

The TCP window size is dynamic and can be adjusted based on factors such as network conditions, congestion, and the receiver's buffer availability.

The size of the window is determined by the formula $\boxed{\text{Window size} = \text{minimum}\ (rwnd, cwnd)}$

- rwnd: receiver window defines the data (number of bytes) that the receiver can accept without making its buffer overflow.

- cwnd: congestion window values determined by the network to avoid congestion. How to get optimal value of cwnd?

Congestion Control:

<u>TCP window: </u>Congestion detection

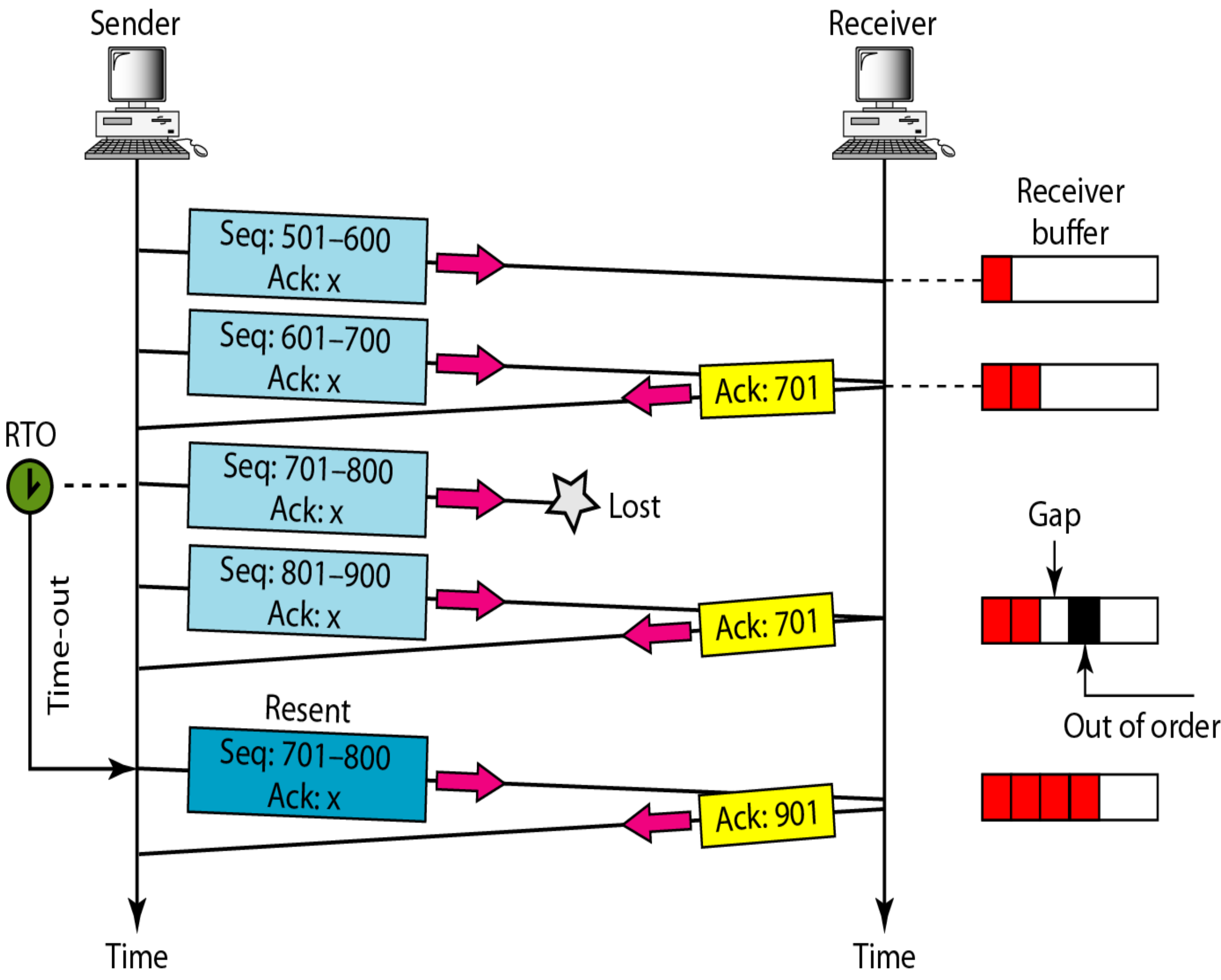
- Before discussing how the value of cwnd should be set and changed, we need to describe how a TCP sender can detect the possible existence of congestion in the network.
- The TCP sender uses the occurrence of two events as signs of congestion in the network:
  - Time-out
  - Receiving three duplicate ACKs.

Congestion
long Queues
waiting time

Congestion Control:

Congestion detection: RTO expires

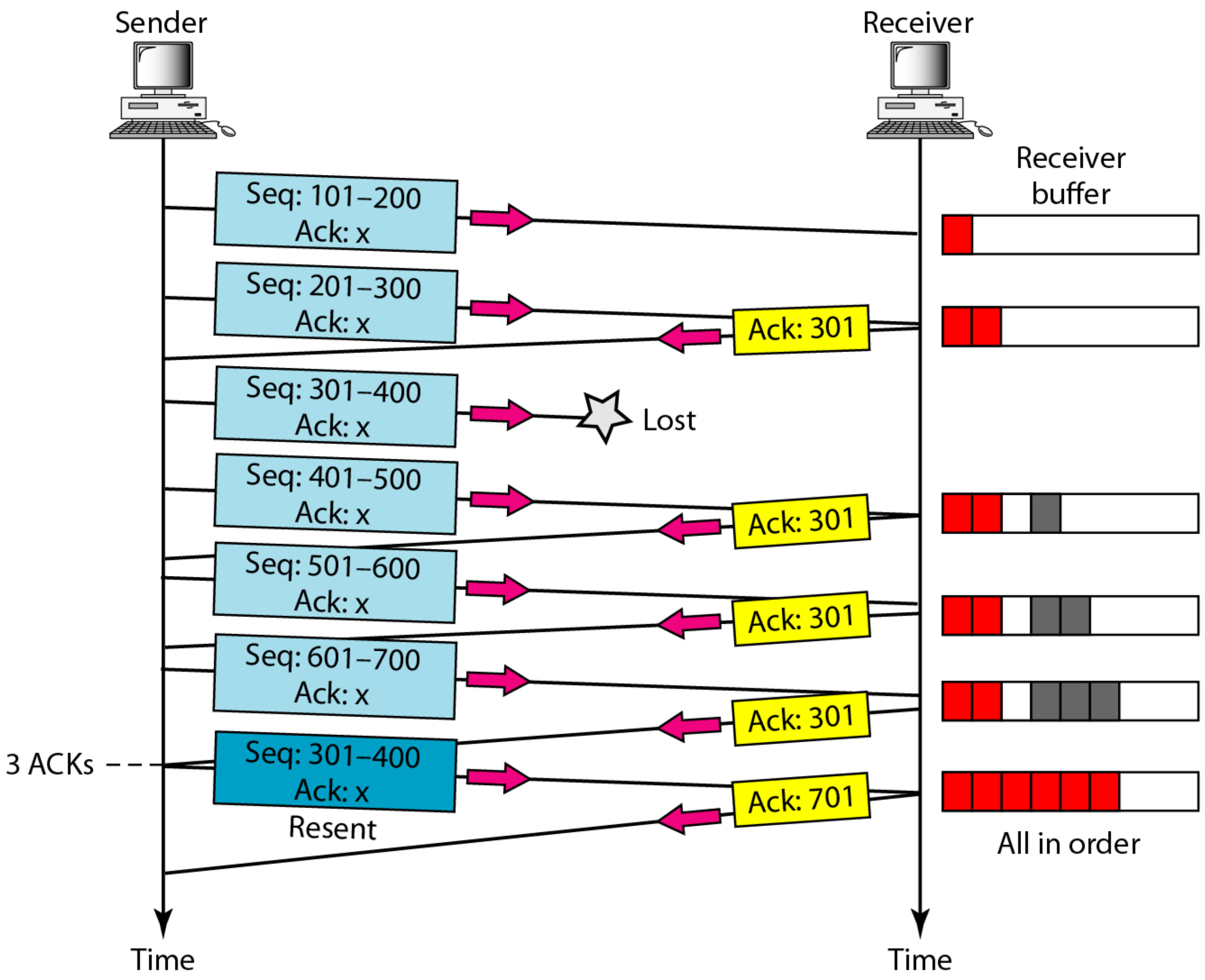
- If a TCP sender does not receive an ACK for a segment or a group of segments before the time-out occurs, it assumes that the corresponding segment or segments are lost and the loss is due to congestion.
- RTO (retransmission timeout) refers to the time interval that TCP waits before retransmitting a packet that has not been acknowledged by the receiver.
- TCP doesn't discard the out-of-order segments but rather stores them and flags them as out-of-order until the missing segments arrive.



Congestion Control:

<u>Congestion detection: 3 Duplicate ACKs</u>

- If the receiver gets out-of-order segments then it sends three duplicate ACKs that trigger the segment transmission.
- This feature of TCP is referred to as “Fast transmission” as in this scheme there is no need to wait till the expiry of time-out.



Congestion Control:

Congestion Control Algorithms

TCP's general policy for handling congestion is based on two algorithms: slow start, and congestion avoidance

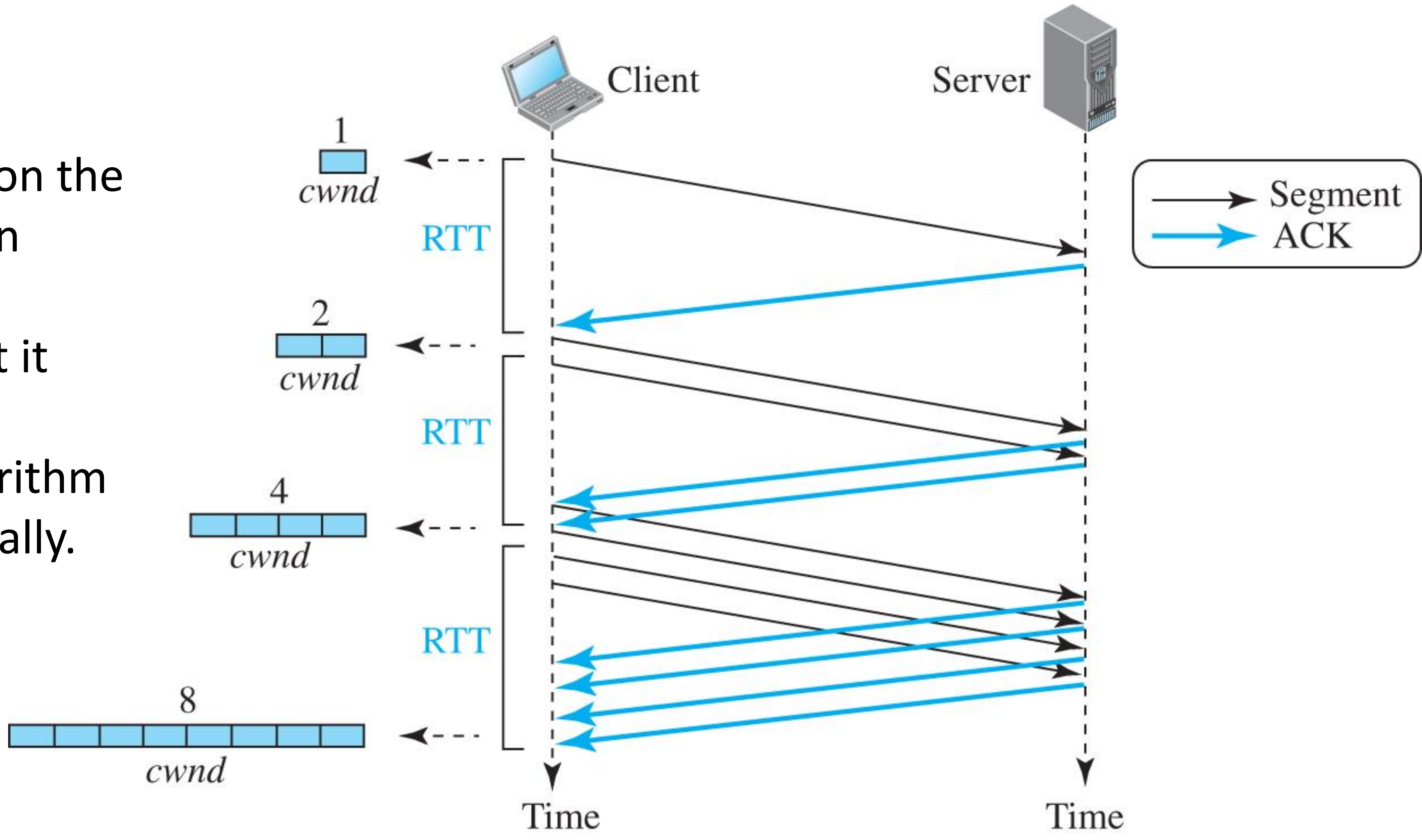
Slow start: In this algorithm, the sender starts sending segments at a very slow rate but increases the segment rate of transmission rapidly till a certain threshold.

- Initially, TCP starts with a small congestion window (**CWND**), often set to one Maximum Segment Size (**MSS**).
- For every acknowledgment (ACK) received, the congestion window size doubles (exponential growth), allowing the sender to increase the transmission rate gradually.
- This continues until a threshold known as **ssthresh** (slow start threshold) is reached.

Congestion Control:

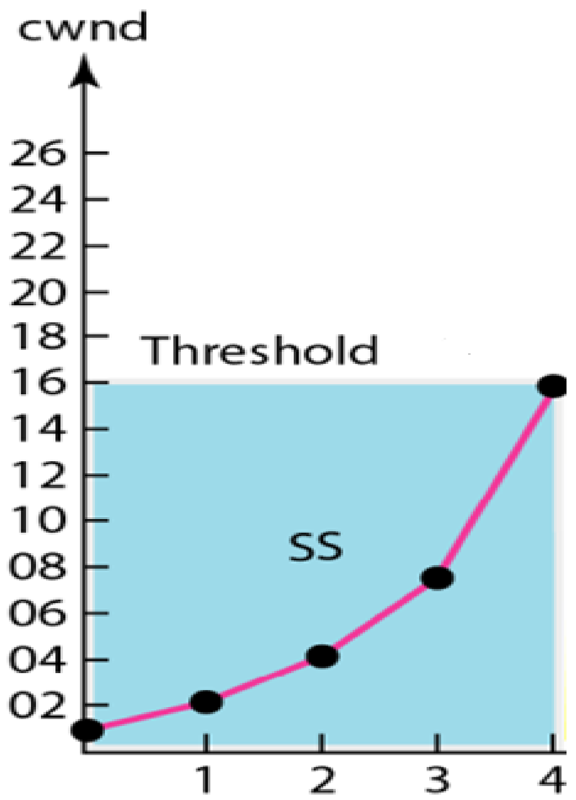
Slow start-exponential increase

The slow-start algorithm is based on the idea that the size of the congestion window (cwnd) starts with one maximum segment size (MSS), but it increases one MSS each time one acknowledgment arrives. the algorithm starts slowly, but grows exponentially.



Congestion Control:

Slow start-exponential increase

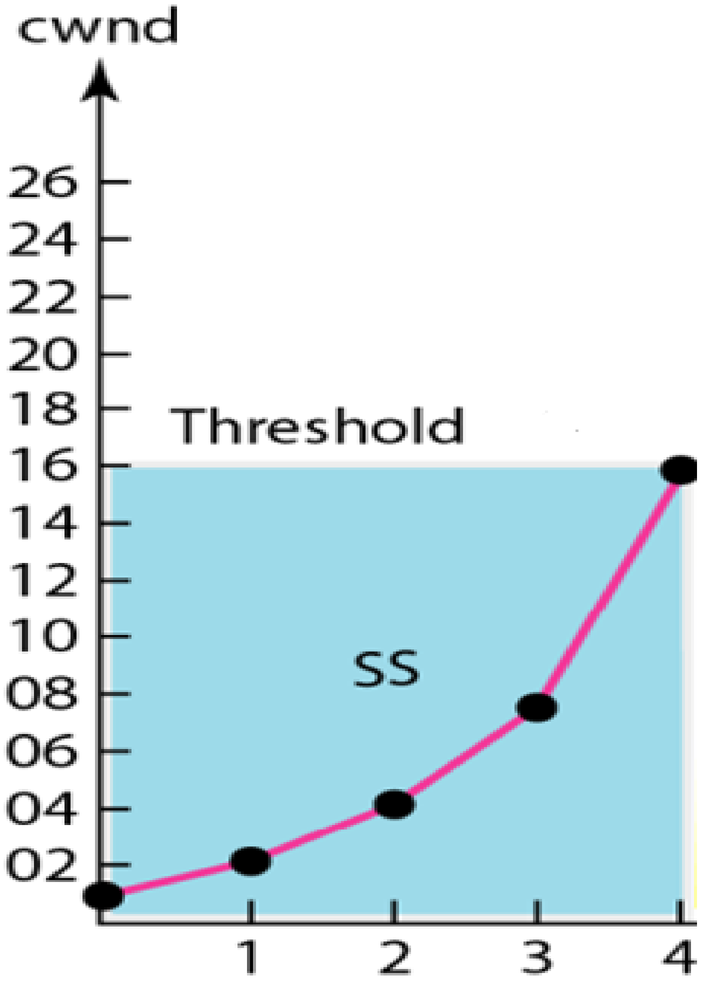


Start	➡	$cwnd = 1$
After round 1	➡	$cwnd = 2^1 = 2$
After round 2	➡	$cwnd = 2^2 = 4$
After round 3	➡	$cwnd = 2^3 = 8$

Congestion Control:

Slow start-exponential increase

In the slow-start algorithm, the size of the congestion window increases exponentially until it reaches a threshold.



Congestion Control:

Congestion Control Algorithms

Congestion avoidance: Once the **ssthresh** threshold is reached, the segment transmission rate is reduced to avoid congestion.

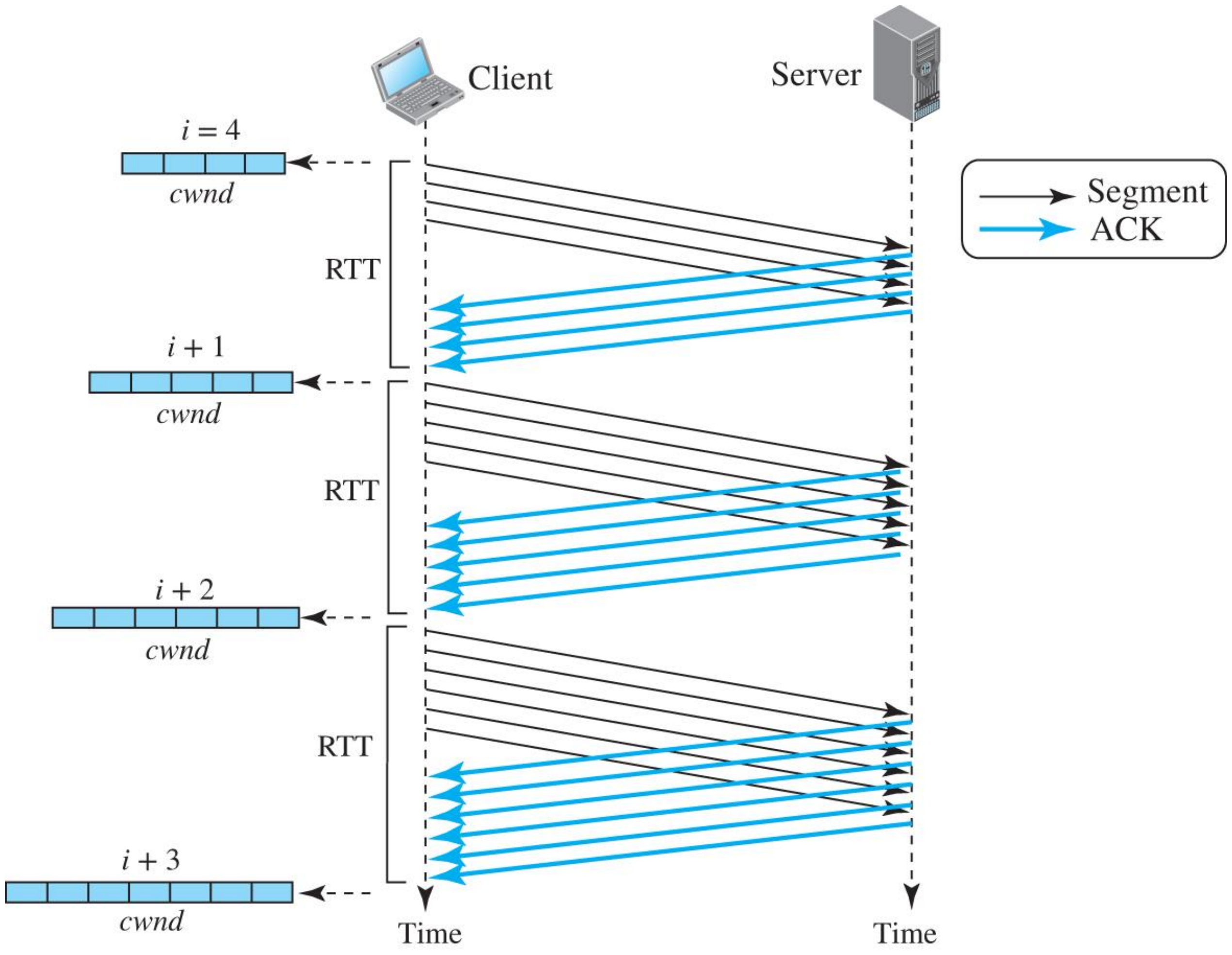
- **ssthresh (Slow Start Threshold)**: The threshold that determines the transition from Slow Start to Congestion Avoidance.

- In this algorithm, the congestion window is increased linearly by one MSS per round-trip time (RTT).

Congestion Control:

Congestion avoidance-linear or additive increase

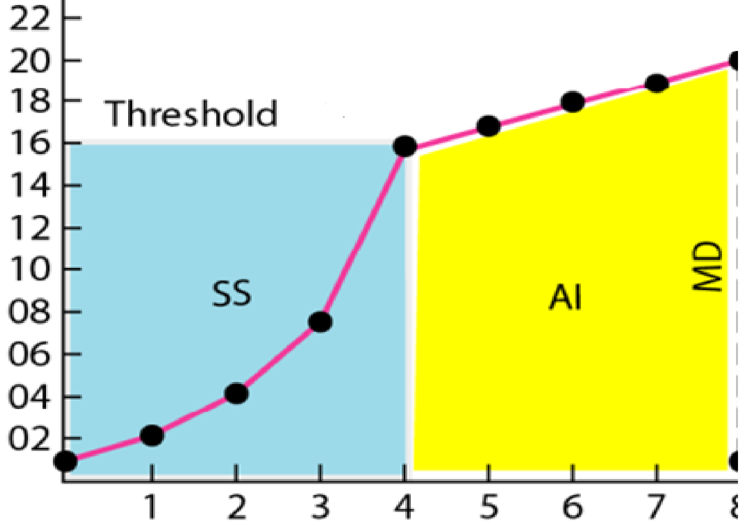
- As in slow start, the window increases exponentially, TCP defines another scheme to avoid congestion.
- In this scheme, cwnd undergoes linear increase instead of exponential.
- When the slow start scheme reaches to the threshold, the Additive scheme begins.
- In this scheme, cwnd increases by 1 in each round.
- The size of the cwnd increases additively till congestion is detected.



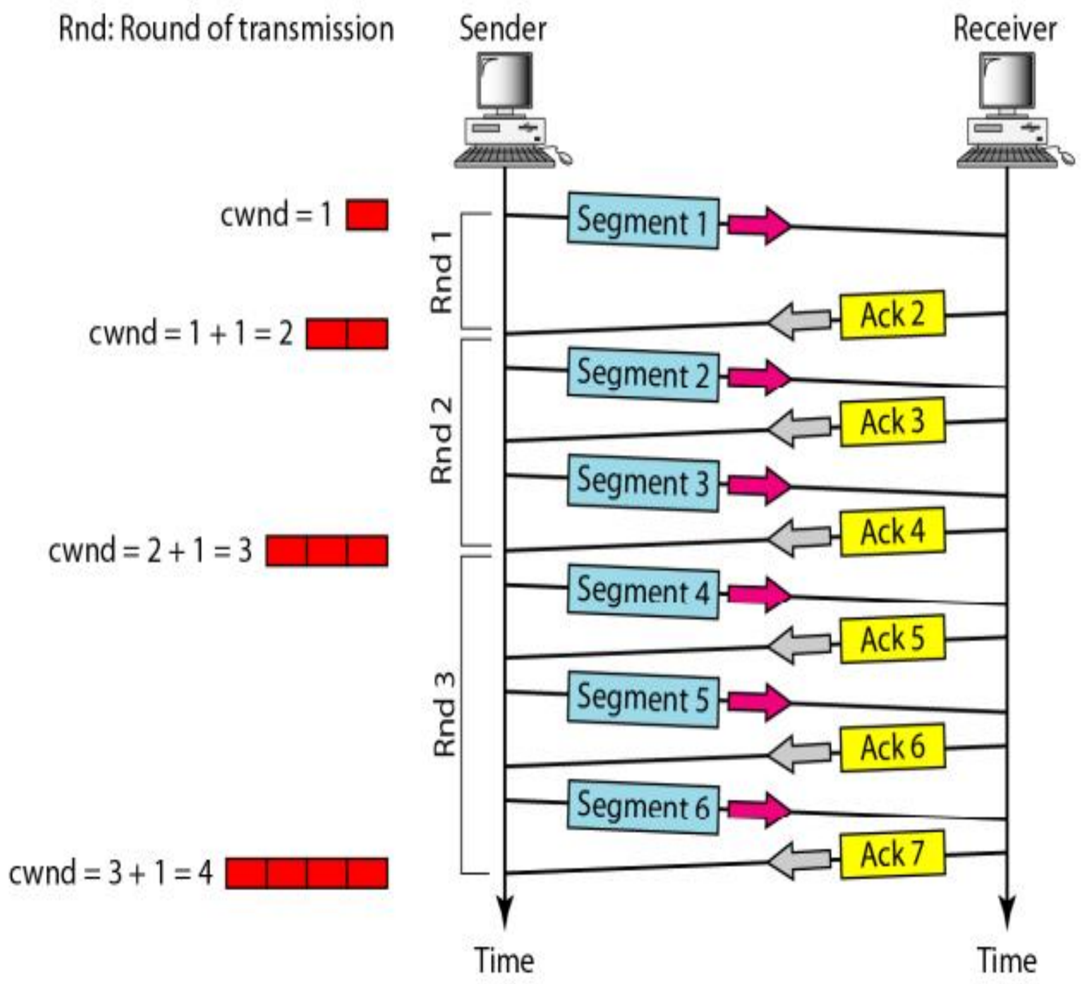
Congestion Control:

Congestion avoidance-linear or additive increase

Start	➡	*cwnd=1*
After round 1	➡	*cwnd=* 1+ 1 =2
After round 2	➡	*cwnd=2+* 1 =3
After round 3	➡	*cwnd=3+* 1 =4



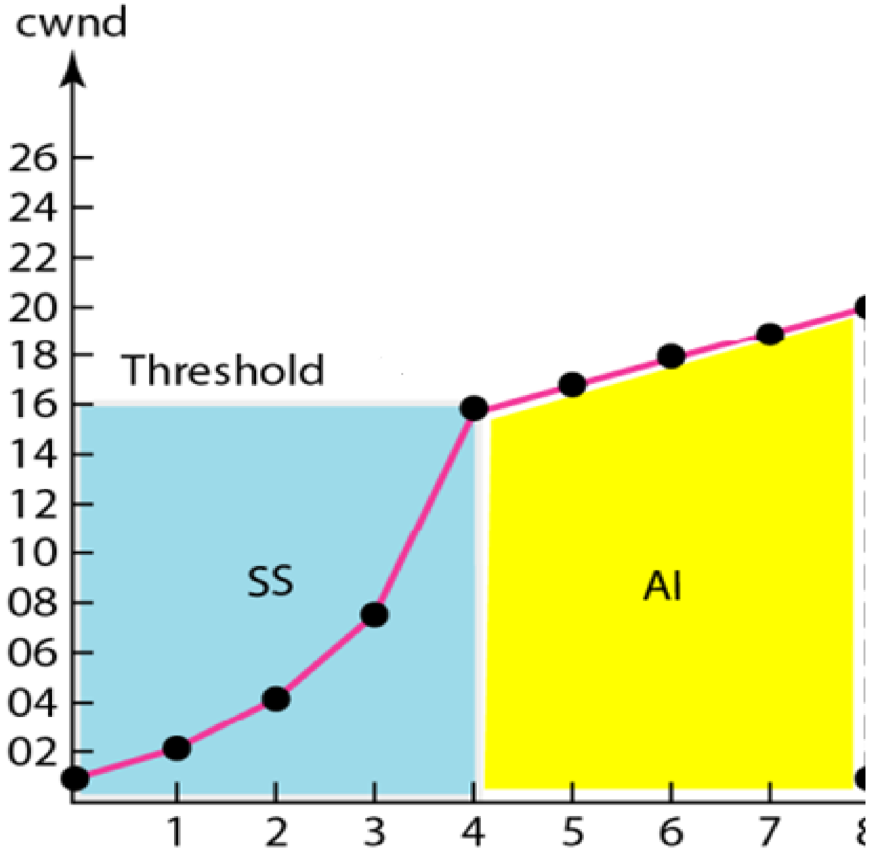
SS: Slow start
AI: Additive increase
MD: Multiplicative decrease



Congestion Control:

Congestion avoidance-linear or additive increase

In the congestion avoidance algorithm, the size of the congestion window increases additively until congestion is detected.



Congestion Control:

<u>Congestion Control Policy</u>

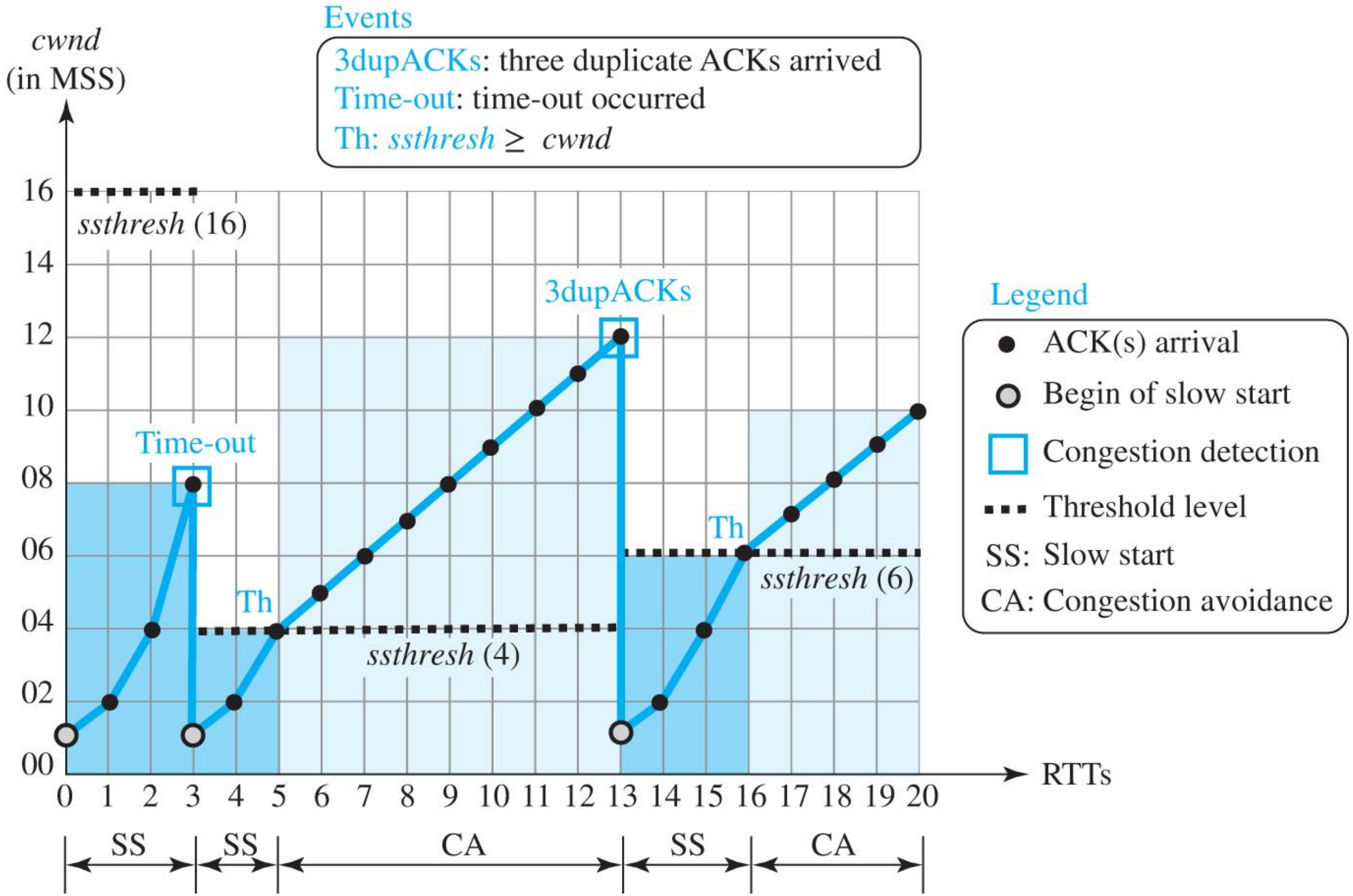
- TCP congestion control is a key component of ensuring reliable data transmission across networks without overwhelming the network.

- The TCP policies**, Taho** and **Additive Increase, Multiplicative Decrease (AIMD)** are closely related concepts that aim to regulate the flow of data to avoid congestion.

Congestion Control:

Taho TCP

I. TCP starts data transfer and sets the ssthresh variable to an ambitious value of 16 MSS.
II. TCP begins at the slow-start (SS) state with the cwnd = 1. The congestion window grows exponentially, but a time-out occurs after the third RTT (before reaching the threshold).
III. TCP assumes that there is congestion in the network. It immediately sets the new ssthresh = 4 MSS (half of the current cwnd, which is 8) and begins a new slow start (SA) state with cwnd = 1 MSS.



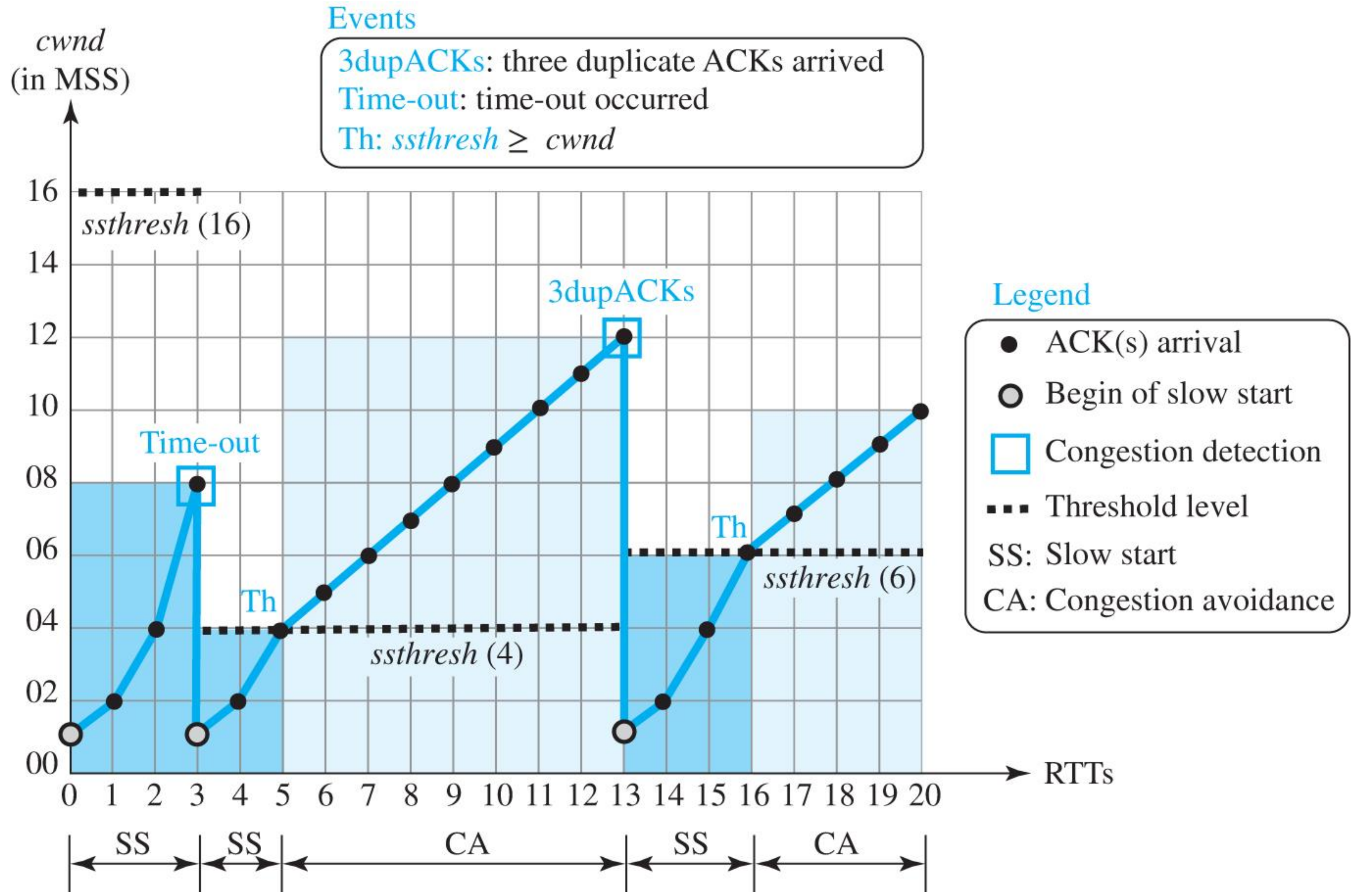
Congestion Control:

Taho TCP

IV. TCP now moves to the congestion avoidance (CA) state and the congestion window grows additively until it reaches cwnd = 12 MSS.

V. At this moment, three duplicate ACKs arrive, another indication of the congestion in the network.

VI. TCP again halves the value of ssthresh to 6 MSS and begins a new slow-start (SS) state. The exponential growth of the cwnd continues

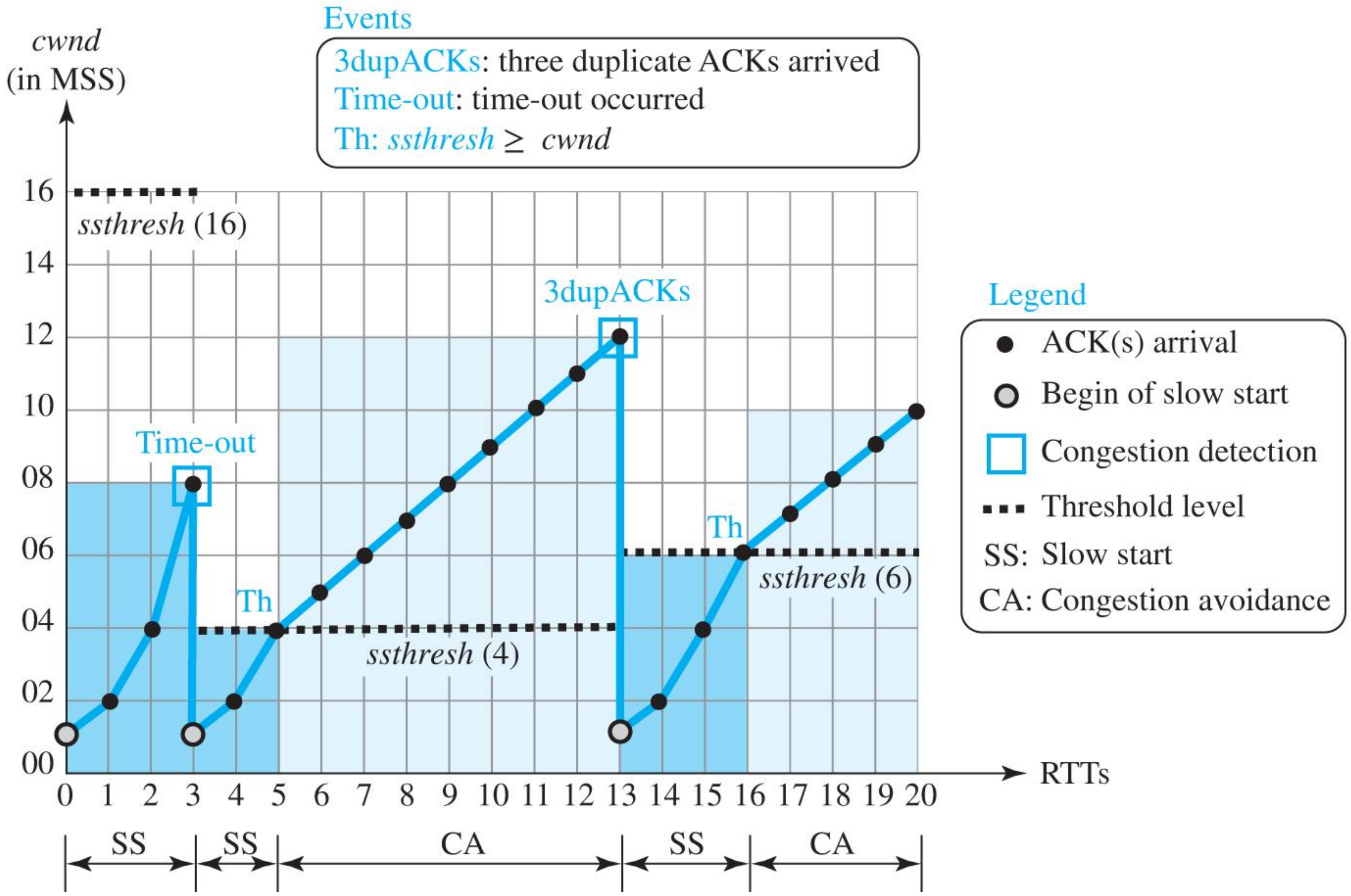


Congestion Control:

Taho TCP

VII. After RTT 15, the size of cwnd is 4 MSS as the size of the window reaches the ssthresh (6) and the TCP moves to the congestion avoidance state.

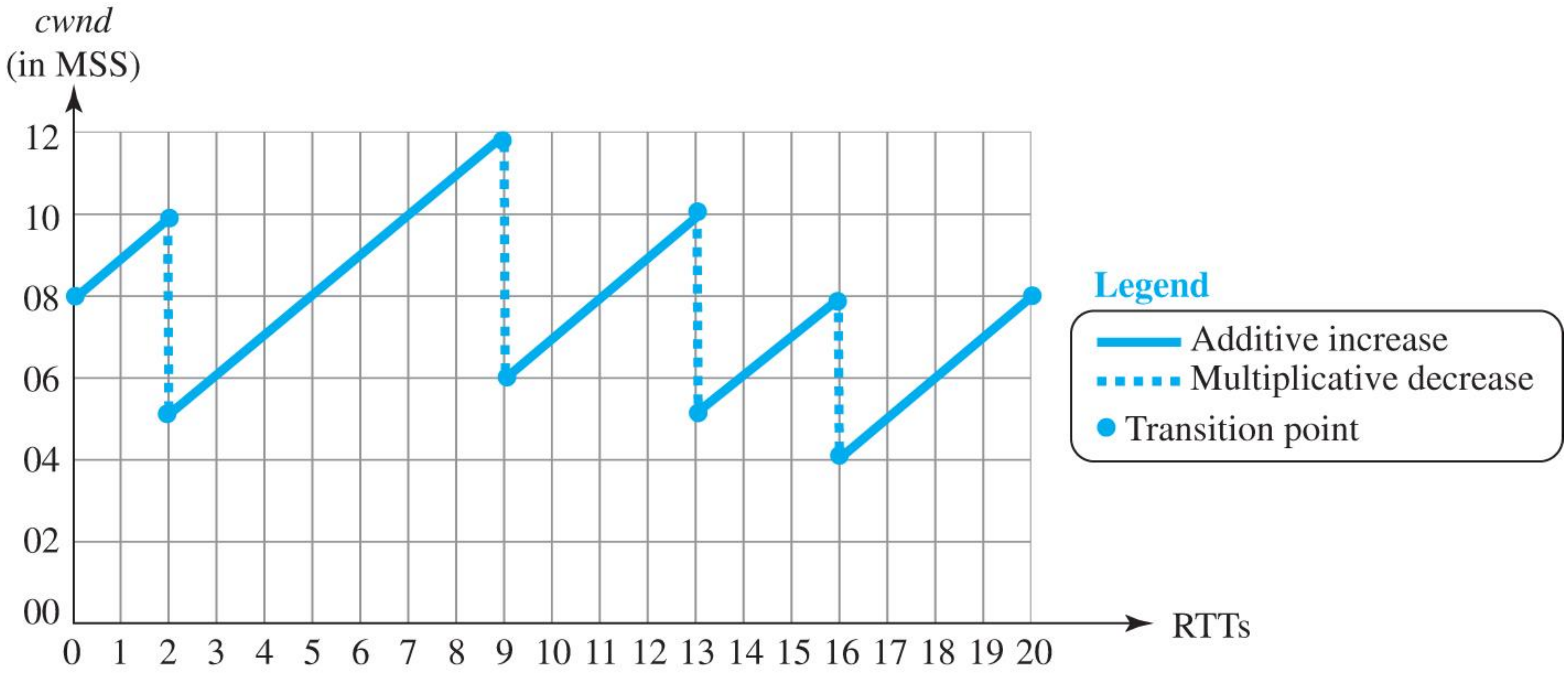
VIII. The data transfer now continues in the congestion avoidance (CA) state until the connection is terminated after RTT 20.



Congestion Control:

Additive increase, multiplicative decrease (AIMD)

The key idea behind **AIMD** is to increase the transmission rate slowly (additively) but decrease it quickly (multiplicatively) to prevent congestion.



Congestion Control:

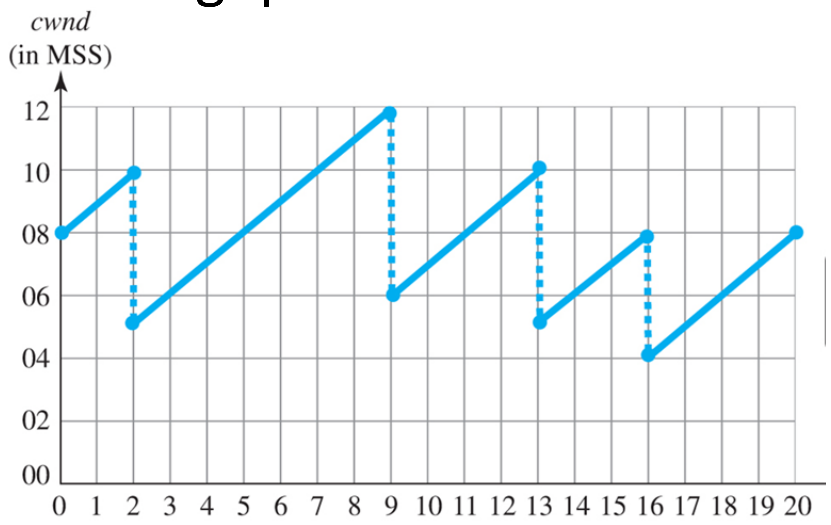
Additive increase, multiplicative decrease (AIMD) - Throughput

The throughput for TCP, which is based on the congestion window behavior, can be found if the cwnd is a constant (flat line) function of RTT.

Throughput = (0.75) Wmax / RTT

If MSS = 10 KB (kilobytes) and RTT = 100 ms , we can calculate the throughput as shown below.

- **Wmax = (10 + 12 + 10 + 8 + 8) / 5 = 9.6 MSS**
- **Throughput = (0.75 Wmax / RTT) = 0.75 * 960 kbps / 100 ms = 7.2 Mbps**



CP: Data Networks Model

Friday: 01/24; 15min for each group,

Last Name	First Name	Group No	Assigned Topic
	Mysha	1	4.1-The need for a protocol architecture
Ali	Sher	1	
Zaidi	Syed Meesam Abbas	1	
Arshad	Aatika	2	4.2- The TCP/IP protocol architecture
Ejaz Akhtar	Nahyan Javed	2	
Haider	Syed Muhammad Aun	2	
Kashif	Hania	3	4.3- The OSI Model
Khowaja	Basil	3	
Mujtaba	Ahmed Abdullah	3	
Rizvi	Syed Zaki Hussain	4	4.4- The Interworking
Sheikh	Ehzem	4	
Anees	Mubashir	4	

Thanks!

Next…

Module 02: Wireless Communication Technology

Chapter 05 (Week No. 04 – 06)

Quiz#1 in Wk # 04