Introduction to TCP using NetSim

Krishna Bharadwaj, Pavithra Krishnan,
Venkatesh Ramaiyan

Department of Electrical Engineering,
Indian Institute of Technology Madras,
Chennai, India.

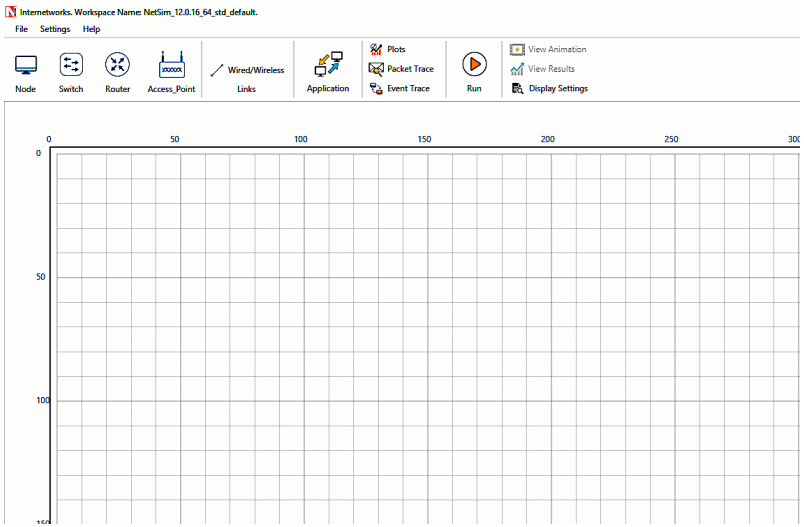
Outline

- About NetSim
- Transport layer protocols
- Transmission control protocol (TCP)
- Features of TCP
  - three way handshake
  - reliable communication
  - congestion control
  - multiplexing and fairness
- Summary
- Suggested exercises
- References

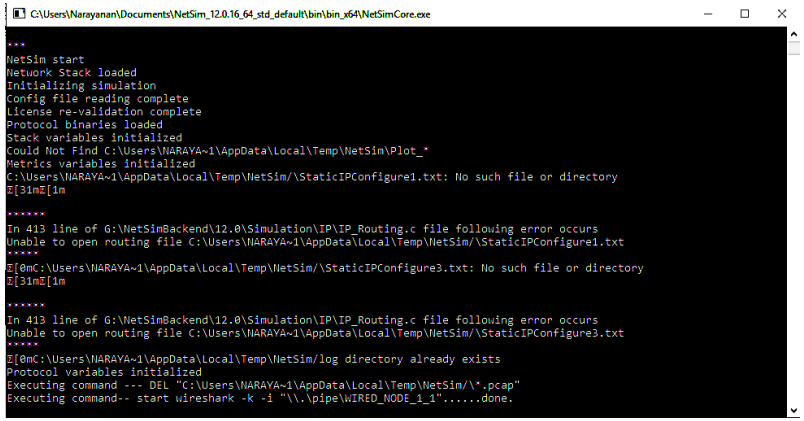
About NetSim

- NetSim - network simulator and emulator
  - end-to-end, full stack, packet level network simulator and emulator
  - enables protocol modeling and simulation
  - used by 400+ customers across 25+ countries
- Highlights
  - technologies: 5G, IoT, SDN, MANETS, VANETS, LTE, Wifi, etc
  - 1000+ nodes
  - GUI and packet animator
  - external interfacing with Matlab, Wireshark and SUMO
- Utility
  - network R&D and academic projects
  - defense applications, tactical communications, public utility networks

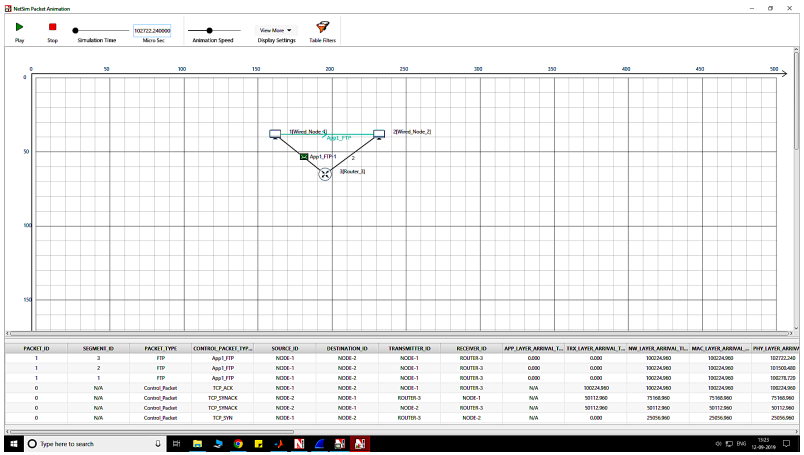
About NetSim



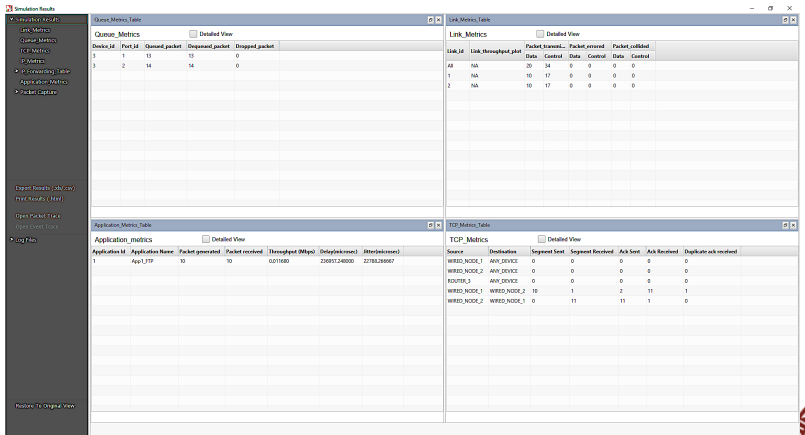
Configure network



Run simulation

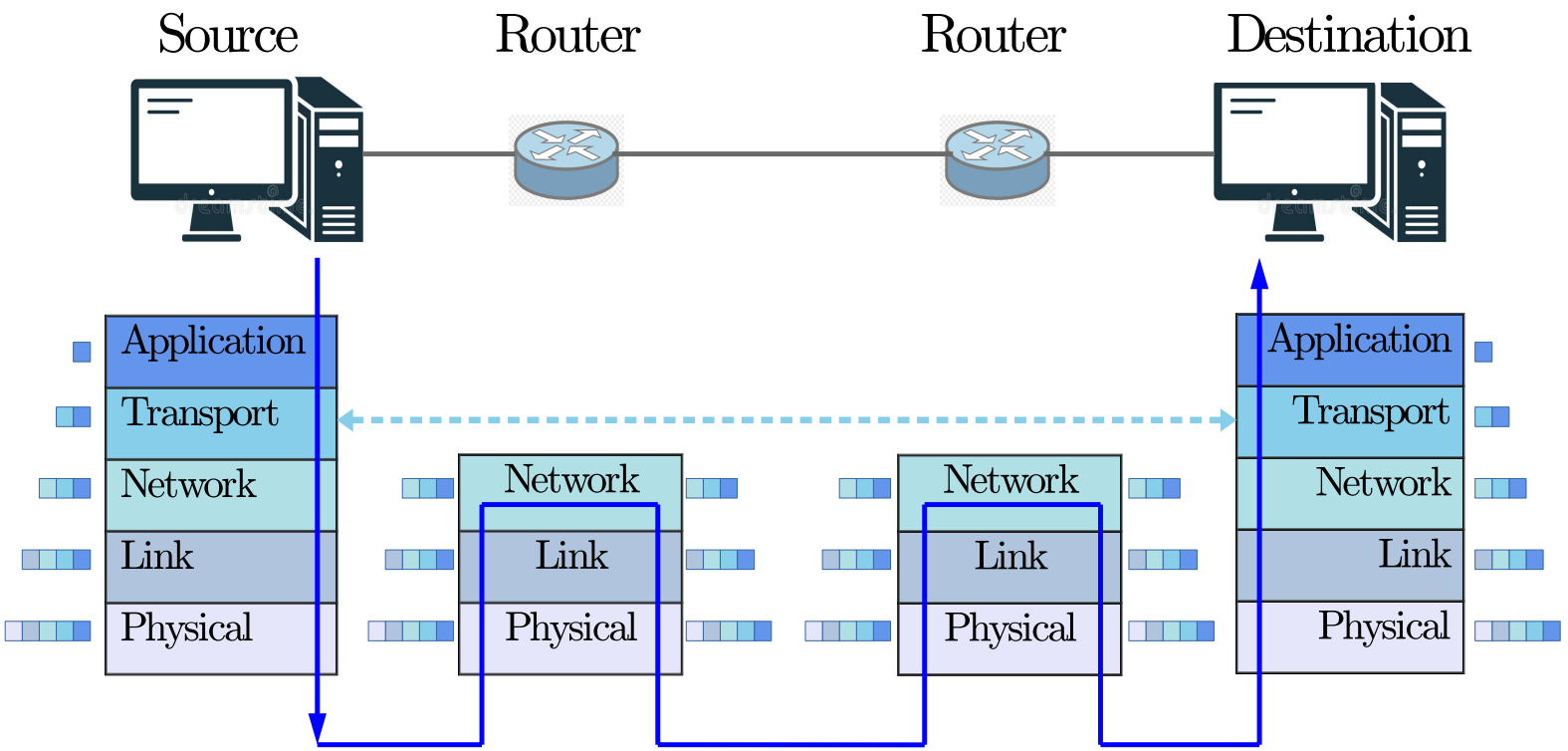


Visualize simulation



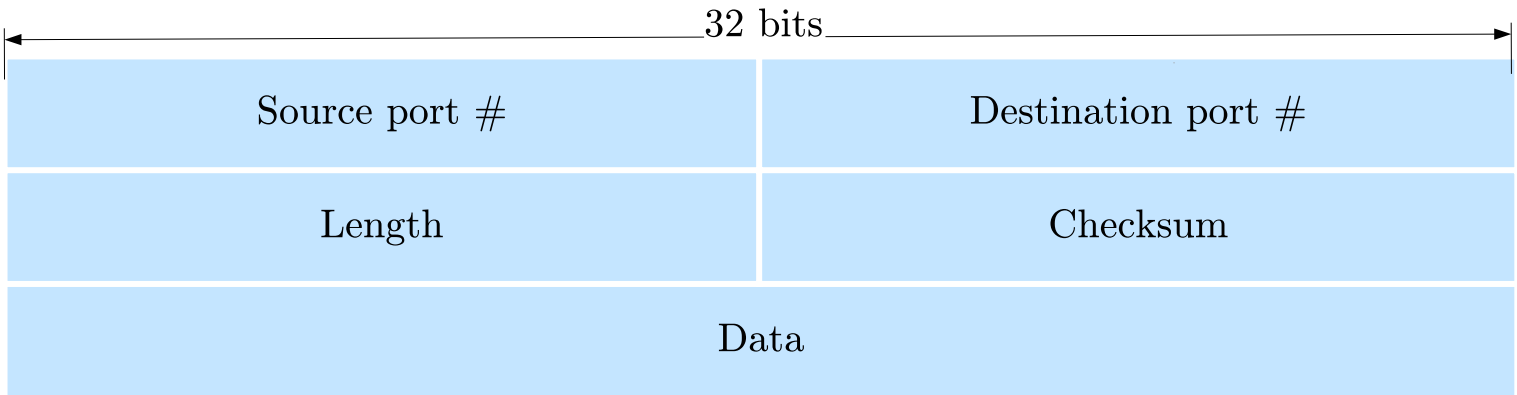
Measurements and metrics

Transport Layer Protocols



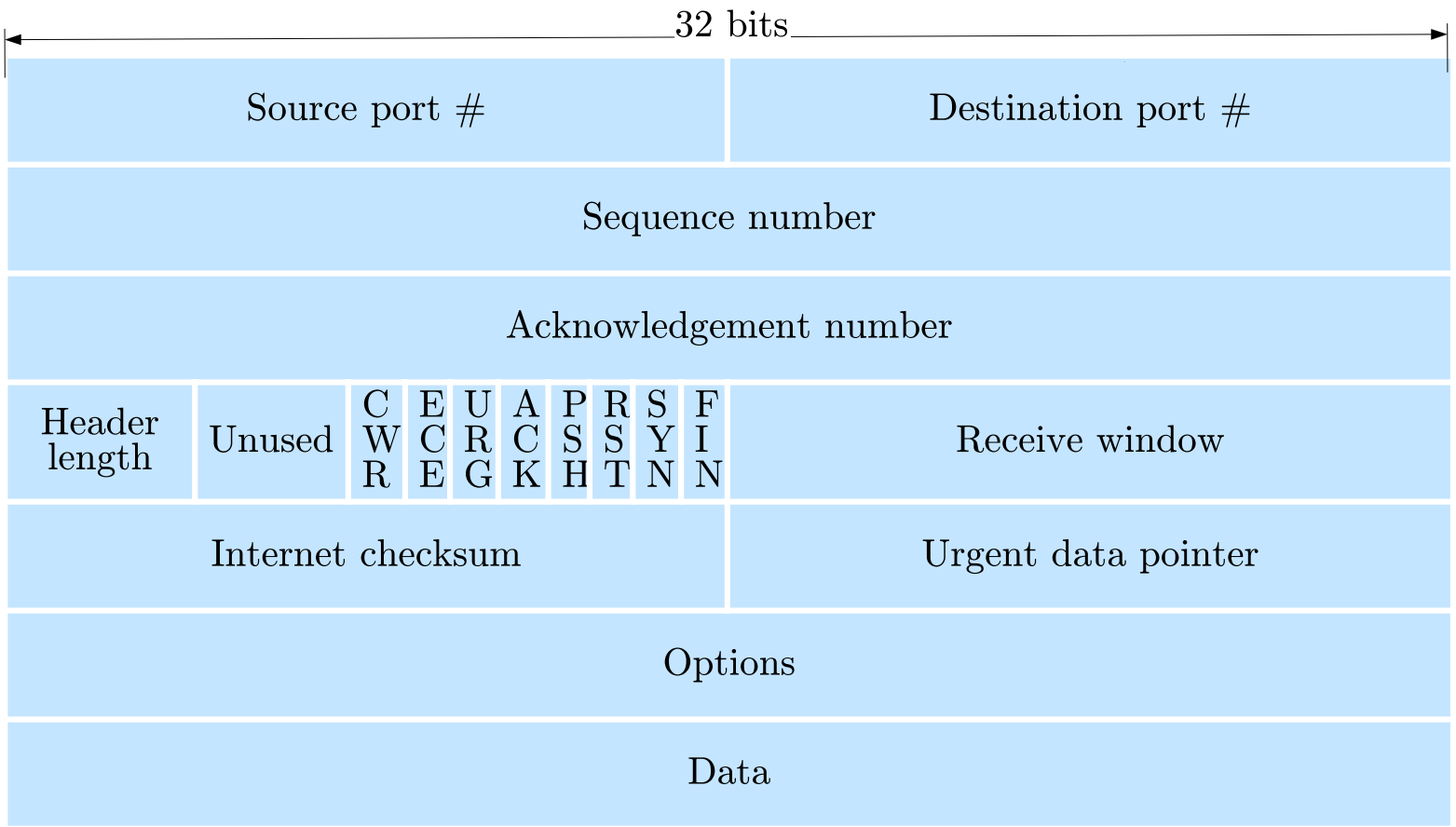
- provides communication services to application processes
- key functions are reliability, multiplexing and congestion control
- popular protocols are TCP, UDP, QUIC

User Datagram Protocol



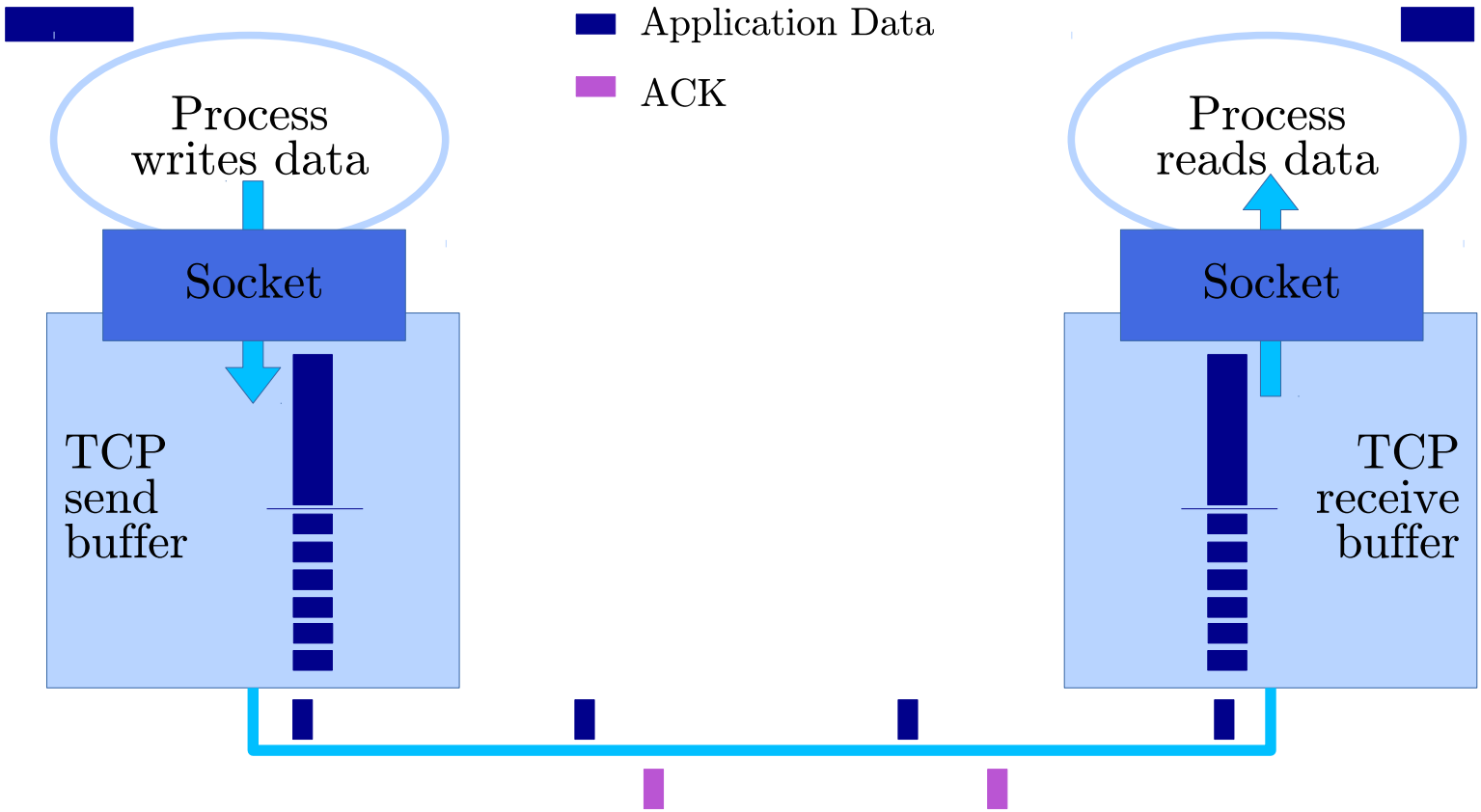
- UDP provides best-effort, connectionless service to applications
- key functions are multiplexing and error detection
- popular applications such as SNMP, DNS, NFS use UDP

Transmission Control Protocol



- TCP provides full-duplex, reliable, connection oriented service
- key functions are multiplexing, flow control and congestion control
- popular applications such as SMTP, Telnet, HTTP, FTP use TCP

Transmission Control Protocol



- TCP converts the data into segments before transmission
- congestion and flow control algorithms paces data transfer

Features of Transmission Control Protocol

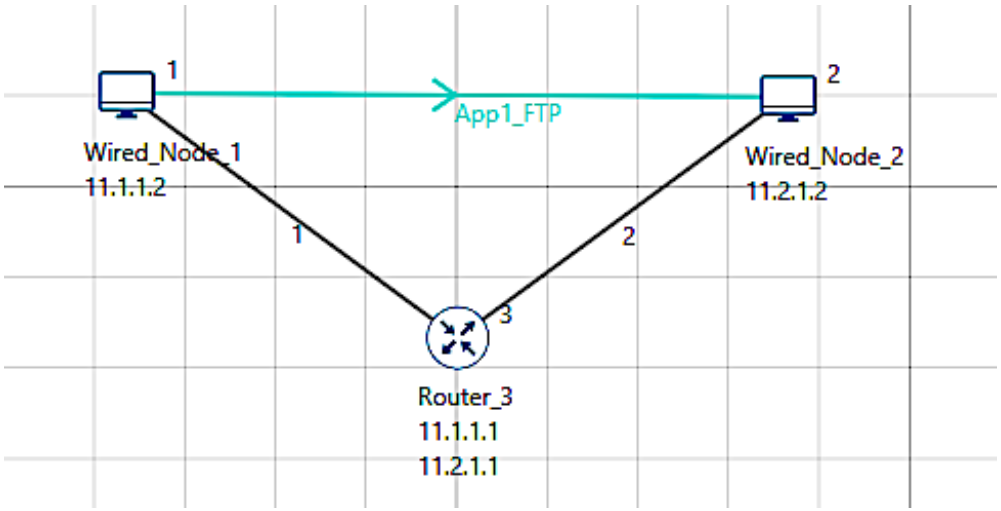
- Connection oriented service
  - connection established before data communication
  - three-way handshake using special control packets
- Reliable communication
  - delivers packets without error and in order
  - uses sequence numbers, acknowledgement, timeout and retransmission
- Flow control
- Congestion control
  - performs end-to-end congestion control
  - uses packet loss and delay to infer congestion and pace traffic
- Multiplexing
  - multiplexing at host using port numbers and IP addresses
  - multiplexing in the network at the routers

Basic TCP Experiments

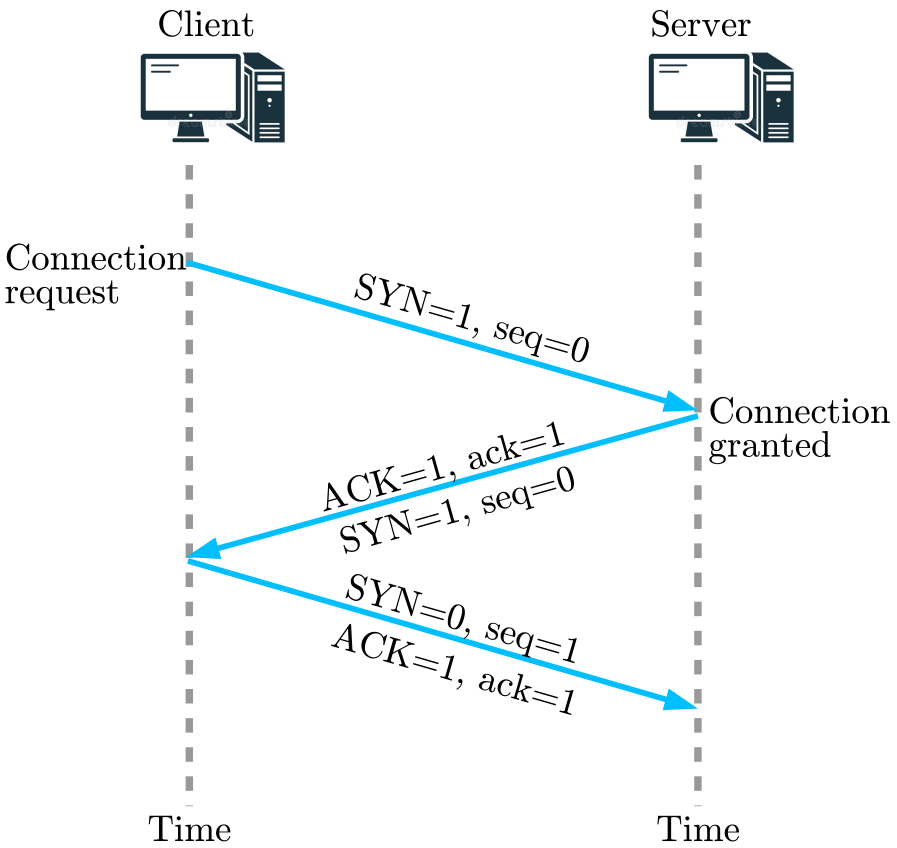
1. Three-way handshake
   - study connection establishment and teardown in TCP
   - focus on three-way handshake, TCP control packet and header
2. Reliable communication
   - study data communication over a lossy link
   - focus on packet sequence numbers, ACKs and retransmissions
3. Congestion control
   - study congestion window evolution with time
   - focus on bandwidth probing, congestion avoidance and throughput
4. Multiplexing in the network
   - study network resource sharing with multiple TCP flows
   - focus on throughput and fairness

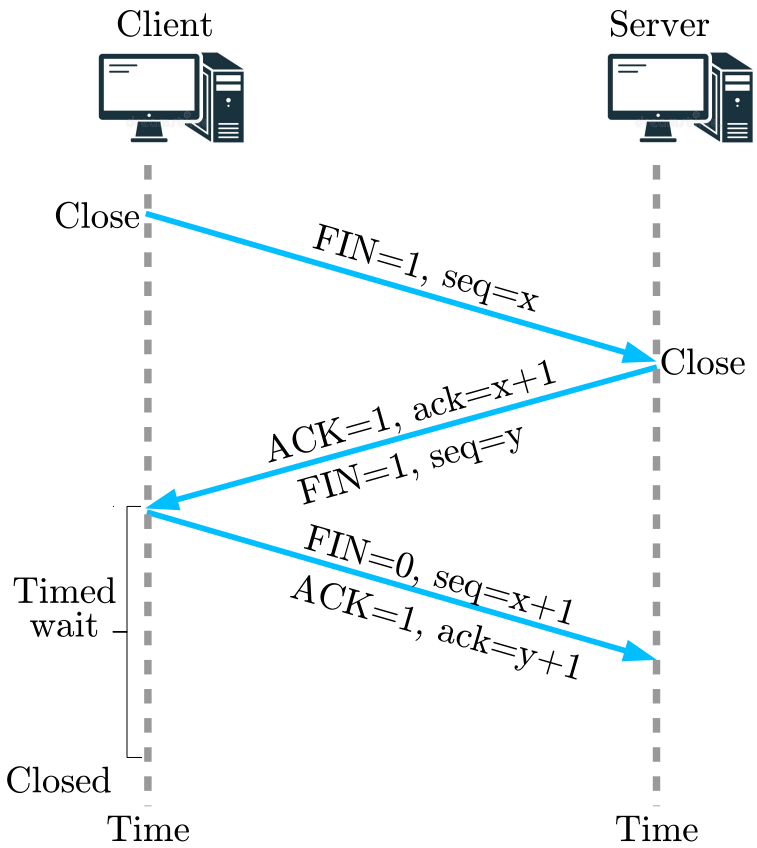
Three-way Handshake (≈ Ex. 4)

- Objective
  - study connection establishment and teardown
  - focus on three-way handshake, TCP control packet and header
- Network setup
  - a client and a server connected by a loss-less link
  - FTP application uses TCP for file transfer



Three-way Handshake





Three-way Handshake: Experiment Configuration

Link Parameters	
Wired link speed	10 Mbps
Wired link BER	0
Wired link propagation delay	25 msecs
Transport Layer Parameters	
Transport Protocol	TCP
Congestion Control	New_Reno
MSS	1460 bytes
Application Parameters	
Application	FTP
File Size	14600 bytes
Miscellaneous	
Simulation Time	20 secs
Wireshark	Enabled On

Three-way Handshake: Results

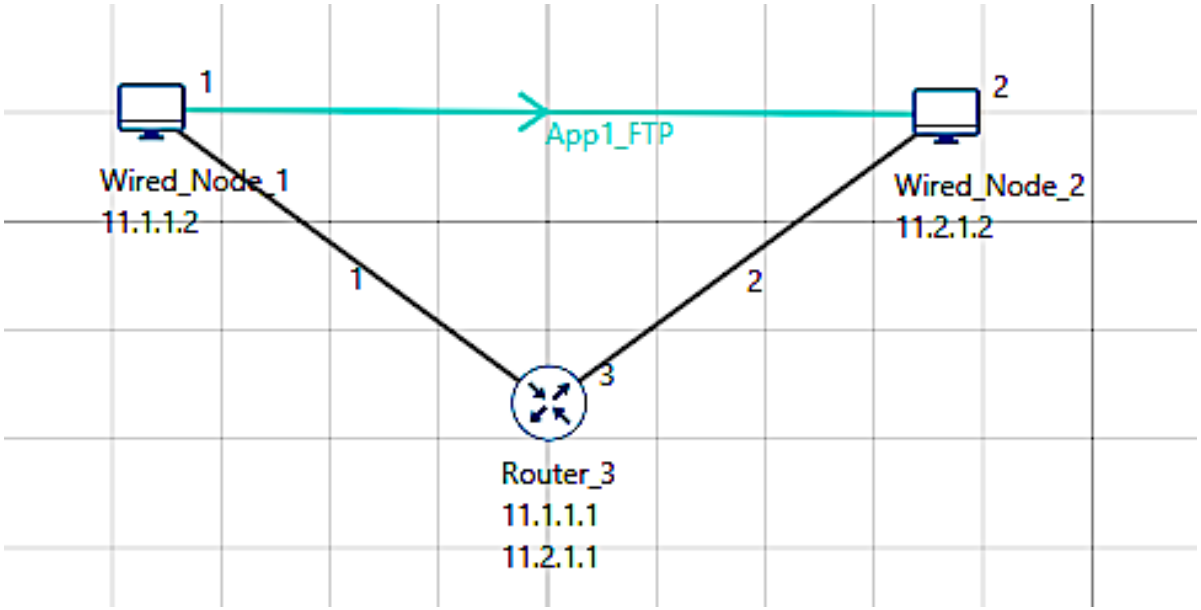
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	0.0.0.0	0.0.0.0	IPv4	20	
2	0.000000	11.1.1.2	11.2.1.2	TCP	44	82 → 36934 [SYN] Seq=0 Win=65535 Len=0 MSS=1460
3	0.100224	11.2.1.2	11.1.1.2	TCP	44	36934 → 82 [SYN, ACK] Seq=0 Ack=1 Win=4380 Len=0 MSS=1460
4	0.100224	11.1.1.2	11.2.1.2	TCP	40	82 → 36934 [ACK] Seq=1 Ack=1 Win=4380 Len=0
5	0.100224	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=1 Win=4380 Len=1460
6	0.100224	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=1461 Win=4380 Len=1460
7	0.100224	11.1.1.2	11.2.1.2	TCP	1500	[TCP Window Full] 82 → 36934 [<None>] Seq=2921 Win=4380 Len=1460
8	0.202825	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=1461 Win=4381 Len=0
9	0.202825	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=4381 Win=5840 Len=1460
10	0.202825	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=5841 Win=5840 Len=1460
11	0.204047	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=2921 Win=4381 Len=0
12	0.204047	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=7301 Win=7300 Len=1460
13	0.204047	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=8761 Win=7300 Len=1460
14	0.205269	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=4381 Win=4381 Len=0
15	0.205269	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=10221 Win=8760 Len=1460
16	0.205269	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=11681 Win=8760 Len=1460
17	0.305373	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=5841 Win=4381 Len=0
18	0.305373	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=13141 Win=10220 Len=1460
19	0.305373	11.1.1.2	11.2.1.2	TCP	40	82 → 36934 [FIN] Seq=14601 Win=10220 Len=0
20	0.306594	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=7301 Win=4381 Len=0
21	0.307816	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=8761 Win=4381 Len=0
22	0.309038	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=10221 Win=4381 Len=0
23	0.310260	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=11681 Win=4381 Len=0
24	0.311481	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=13141 Win=4381 Len=0
25	0.407920	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=14601 Win=4381 Len=0
26	0.407974	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [FIN, ACK] Seq=1 Ack=14601 Win=4381 Len=0
27	0.408027	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [FIN] Seq=2 Win=4381 Len=0
28	0.408027	11.1.1.2	11.2.1.2	TCP	40	82 → 36934 [ACK] Seq=14602 Ack=3 Win=18980 Len=0

Three-way Handshake: Inferences

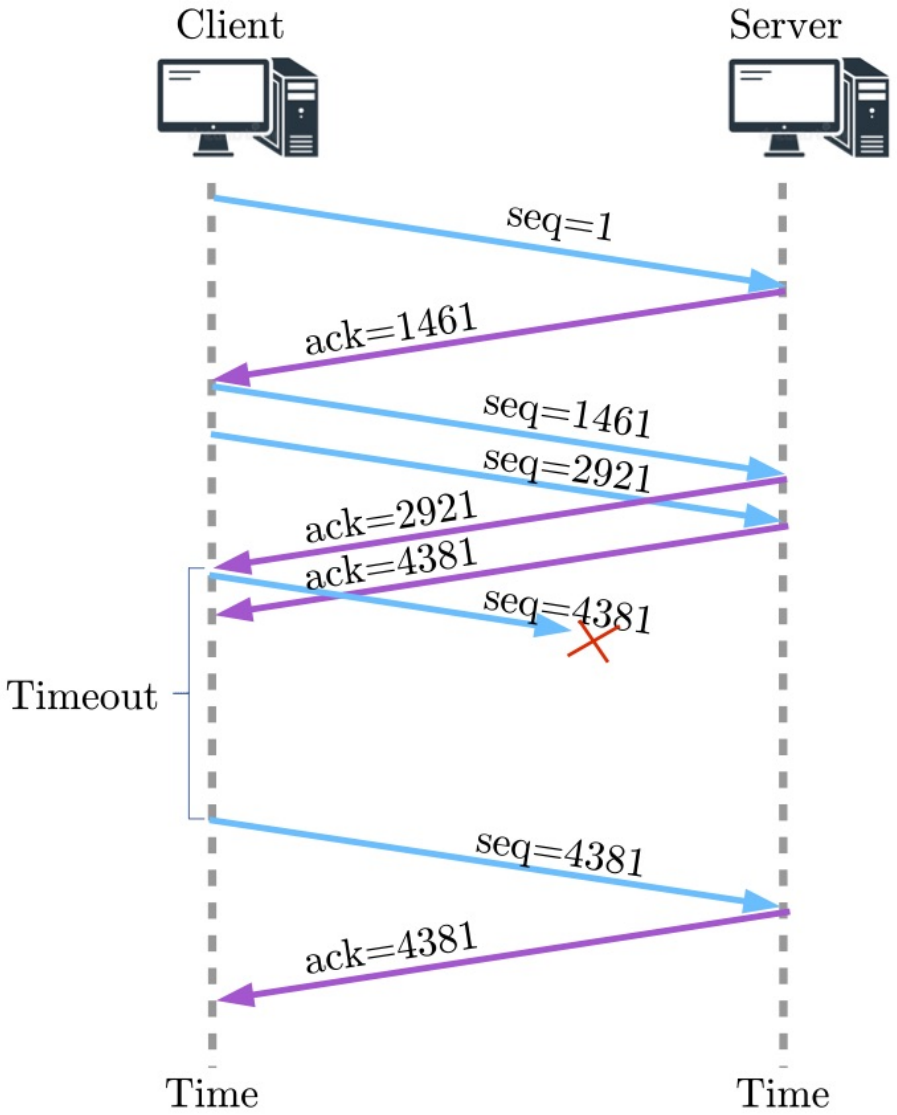
- Connection oriented service
  - data communication after three-way handshake
  - connection terminated with three-way handshake as well
  - TCP header carries information for handshake
- NetSim simulator
  - handshake observed in animator, PCAP file and packet trace
  - PCAP file permits easy review of the TCP header

Reliable Communication

- Objective
  - study data communication over a lossy link, with TCP
  - focus on fragmentation, sequence numbers, ACKs and retransmission
- Network setup
  - a client and a server connected by a lossy link
  - FTP application uses TCP for file transfer



Reliable Communication



Reliable Communication: Experiment Configuration

Link Parameters	
Wired link speed	10 Mbps
Wired link BER	1e-5
Wired link propagation delay	25 msecs
Transport Layer Parameters	
Transport Protocol	TCP
Congestion Control	New_Reno
MSS	1460 bytes
Application Parameters	
Application	FTP
File Size	14600 bytes
Miscellaneous	
Simulation Time	20 secs
Wireshark	Enabled On

Reliable Communication: Results

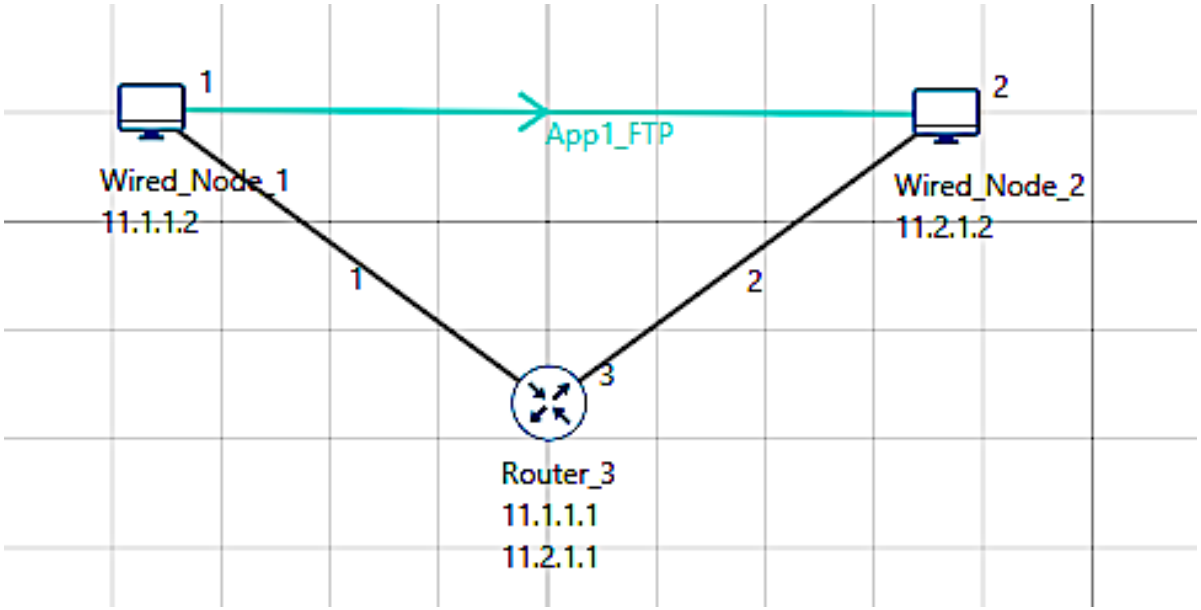
	Time	Source	Destination	Protocol	Length	Info
1	0.000000	0.0.0.0	0.0.0.0	IPv4	20	
2	0.050112	11.1.1.2	11.2.1.2	TCP	44	82 → 36934 [SYN] Seq=0 Win=32767 Len=0 MSS=1460
3	0.050112	11.2.1.2	11.1.1.2	TCP	44	36934 → 82 [SYN, ACK] Seq=0 Ack=1 Win=4380 Len=0 MSS=1460
4	0.150330	11.1.1.2	11.2.1.2	TCP	40	82 → 36934 [ACK] Seq=1 Ack=1 Win=4380 Len=0
5	0.152720	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=1 Win=4380 Len=1460
6	0.152720	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=1461 Win=4381 Len=0
7	0.153942	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=1461 Win=4380 Len=1460
8	0.153942	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=2921 Win=4381 Len=0
9	0.155163	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=2921 Win=4380 Len=1460
10	0.155163	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=4381 Win=4381 Len=0
11	0.256489	11.1.1.2	11.2.1.2	TCP	1500	[TCP Previous segment not captured] 82 → 36934 [<None>] Seq=5841 Win=58
12	0.256489	11.2.1.2	11.1.1.2	TCP	40	[TCP Dup ACK 10#1] 36934 → 82 [ACK] Seq=1 Ack=4381 Win=4381 Len=0
13	0.257711	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=7301 Win=7300 Len=1460
14	0.257711	11.2.1.2	11.1.1.2	TCP	40	[TCP Dup ACK 10#2] 36934 → 82 [ACK] Seq=1 Ack=4381 Win=4381 Len=0
15	0.258932	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=8761 Win=7300 Len=1460
16	0.258932	11.2.1.2	11.1.1.2	TCP	40	[TCP Dup ACK 10#3] 36934 → 82 [ACK] Seq=1 Ack=4381 Win=4381 Len=0
17	0.260154	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=10221 Win=8760 Len=1460
18	0.260154	11.2.1.2	11.1.1.2	TCP	40	[TCP Dup ACK 10#4] 36934 → 82 [ACK] Seq=1 Ack=4381 Win=4381 Len=0
19	0.261376	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=11681 Win=8760 Len=1460
20	0.261376	11.2.1.2	11.1.1.2	TCP	40	[TCP Dup ACK 10#5] 36934 → 82 [ACK] Seq=1 Ack=4381 Win=4381 Len=0
21	0.361479	11.1.1.2	11.2.1.2	TCP	1500	[TCP Out-Of-Order] 82 → 36934 [<None>] Seq=4381 Win=5840 Len=1460
22	0.361479	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=13141 Win=4381 Len=0
23	0.464027	11.1.1.2	11.2.1.2	TCP	1500	82 → 36934 [<None>] Seq=13141 Win=4380 Len=1460
24	0.464027	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [ACK] Seq=1 Ack=14601 Win=4381 Len=0
25	0.464080	11.1.1.2	11.2.1.2	TCP	40	82 → 36934 [FIN] Seq=14601 Win=4380 Len=0
26	0.464080	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [FIN, ACK] Seq=1 Ack=14601 Win=4381 Len=0
27	0.464080	11.2.1.2	11.1.1.2	TCP	40	36934 → 82 [FIN] Seq=2 Win=4381 Len=0
28	0.564345	11.1.1.2	11.2.1.2	TCP	40	82 → 36934 [ACK] Seq=14602 Ack=3 Win=5840 Len=0

Reliable Communication: Inference

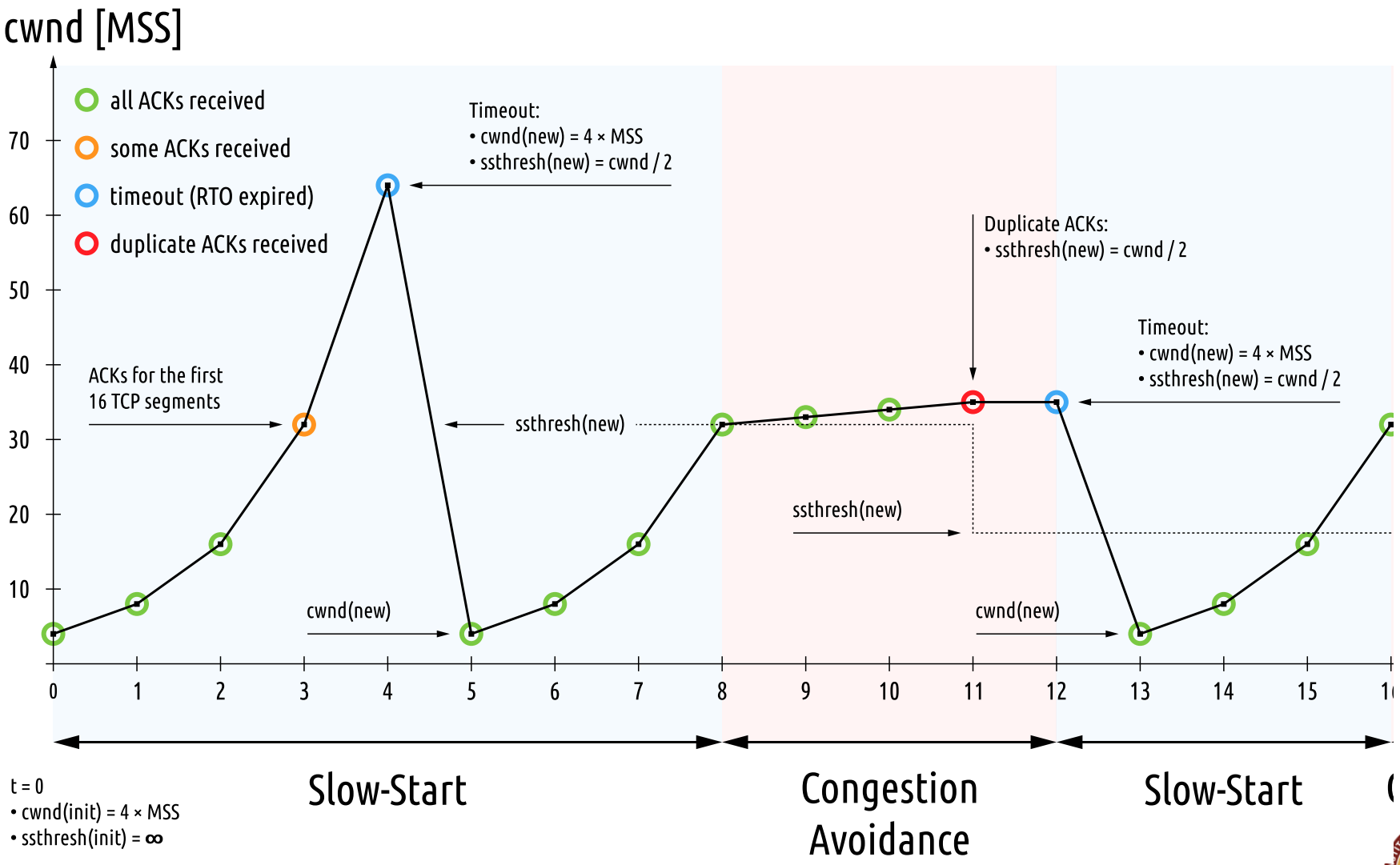
- Reliable communication
  - file transmitted as multiple segments, with TCP headers
  - sequence numbers help track bytes transmitted and received
  - packet loss can occur due to channel errors or congestion
  - timeout and duplicate acknowledgements help detect packet loss
  - connection terminated after the file is transferred
- NetSim simulator
  - data transfer observed in animator, PCAP file and packet trace
  - PCAP file permits easy review of the TCP flow and header

TCP Congestion Control (≈ Ex. 8, 5)

- Objective
  - study congestion window evolution with time
  - focus on bandwidth probing, congestion avoidance and throughput
- Network setup
  - a client and a server connected by a lossy link
  - FTP application seeks to transfer a very large file using TCP



TCP Congestion Control

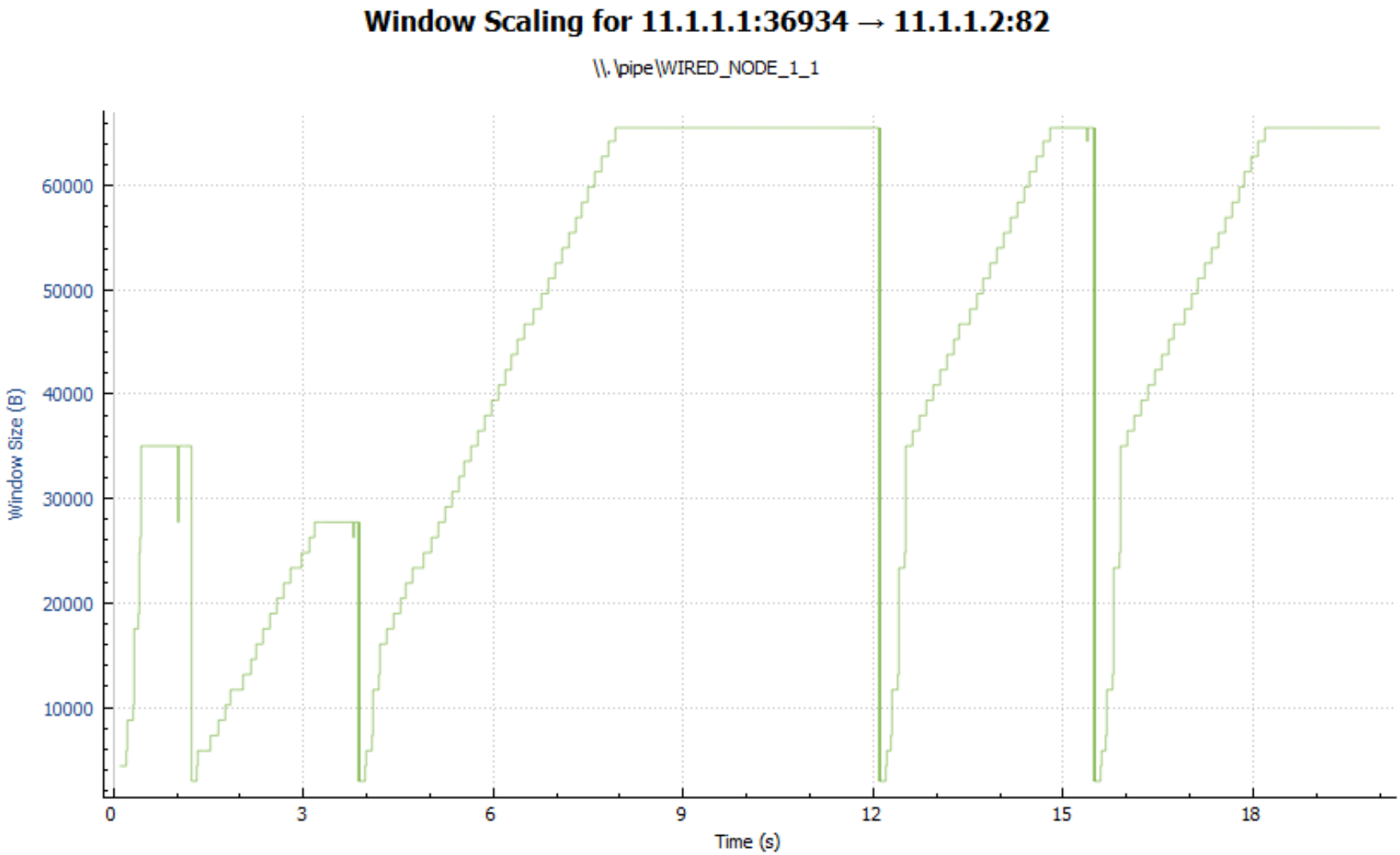


source: https://witestlab.poly.edu/blog/

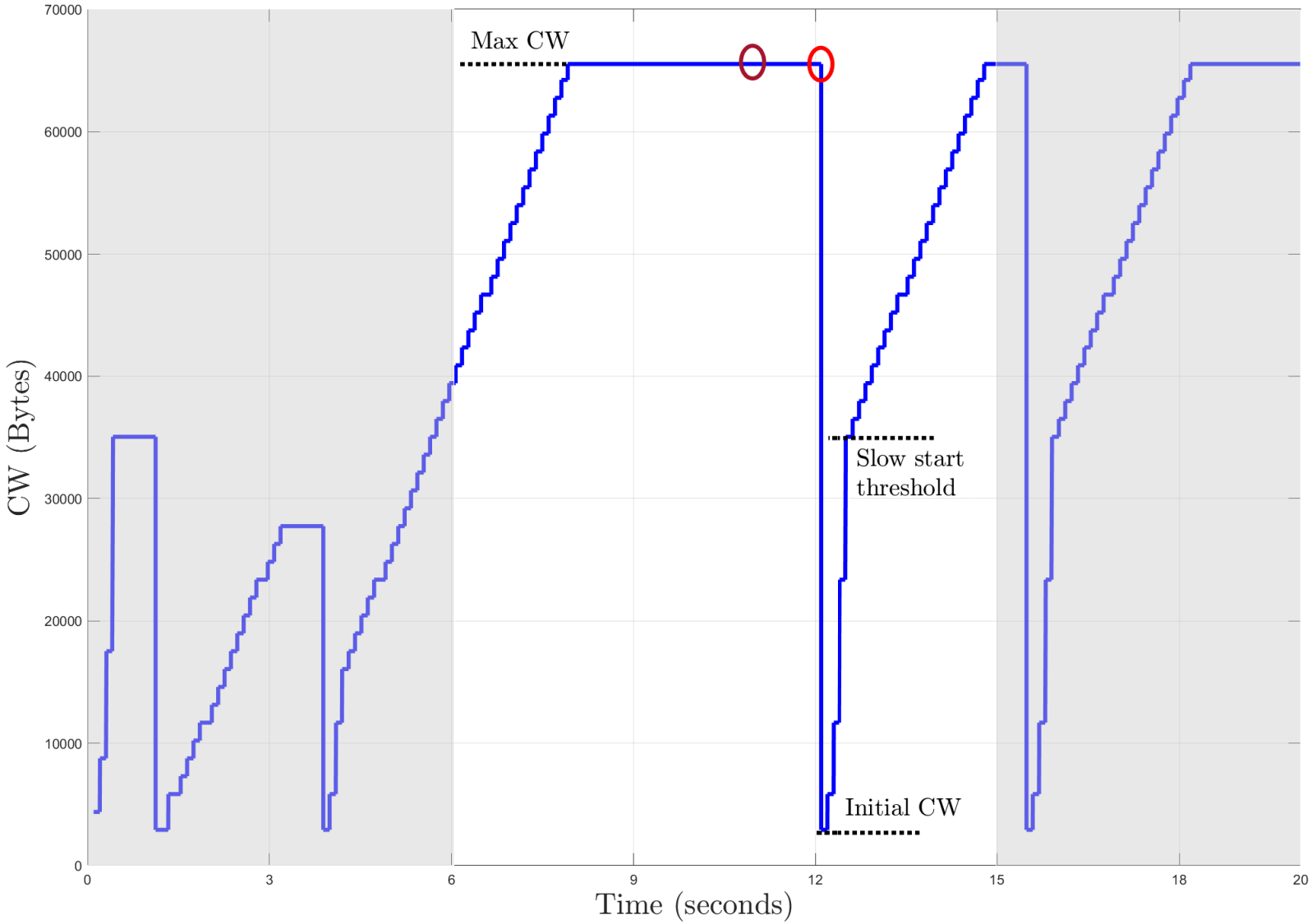
TCP Congestion Control: Experiment Configuration

Link Parameters	
Wired link speed	10 Mbps
Wired link BER	1e-7
Wired link propagation delay	25 msecs
Transport Layer Parameters	
Transport Protocol	TCP
Congestion Control	Old_Tahoe, Tahoe, New_Reno
MSS	1460
Application Parameters	
Application	FTP
File Size	10^8 Bytes
Miscellaneous	
Simulation Time	20 secs
Wireshark	Enabled On

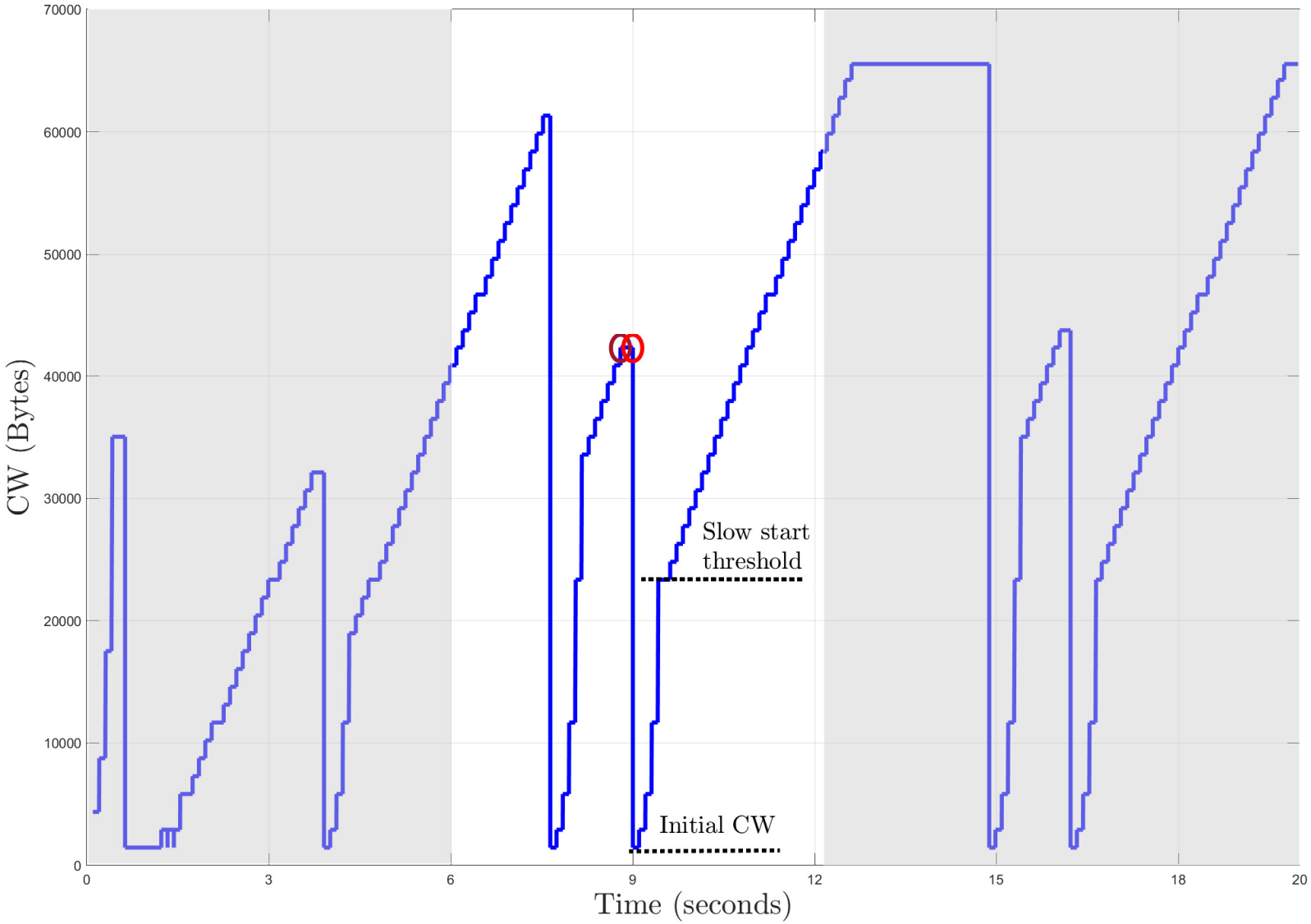
TCP Congestion Control: Old Tahoe



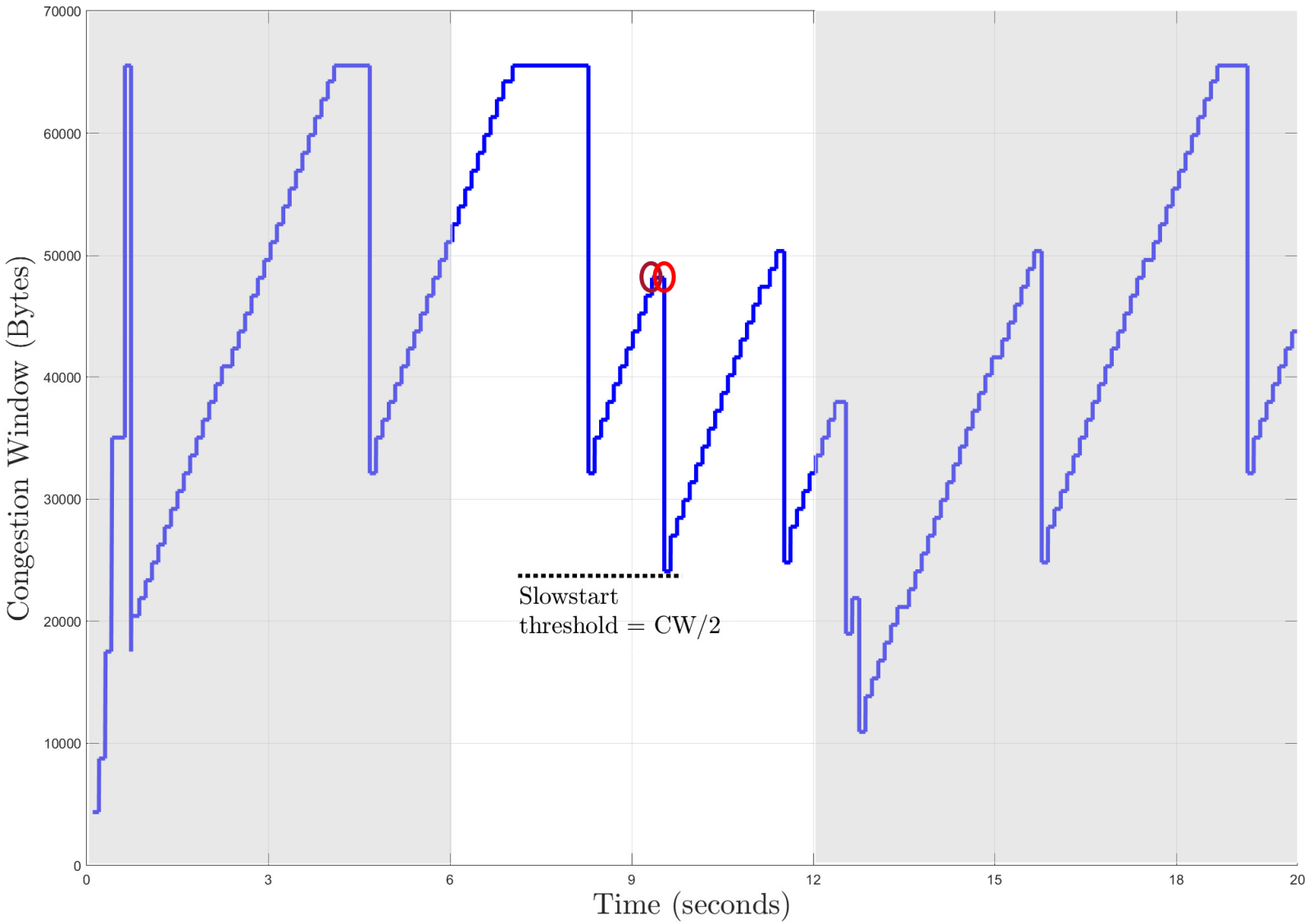
TCP Congestion Control: Old Tahoe



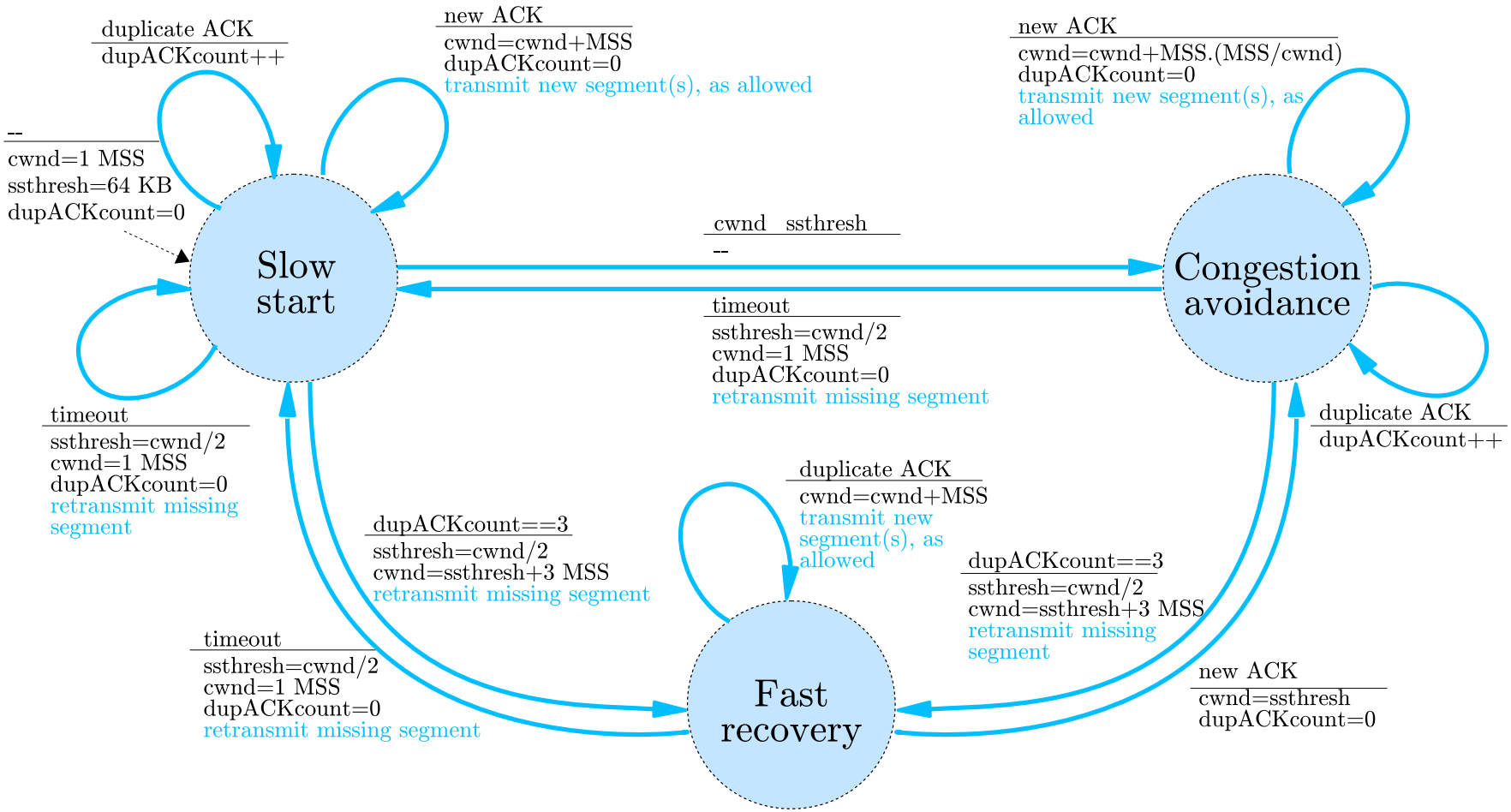
TCP Congestion Control: Tahoe



TCP Congestion Control: New Reno

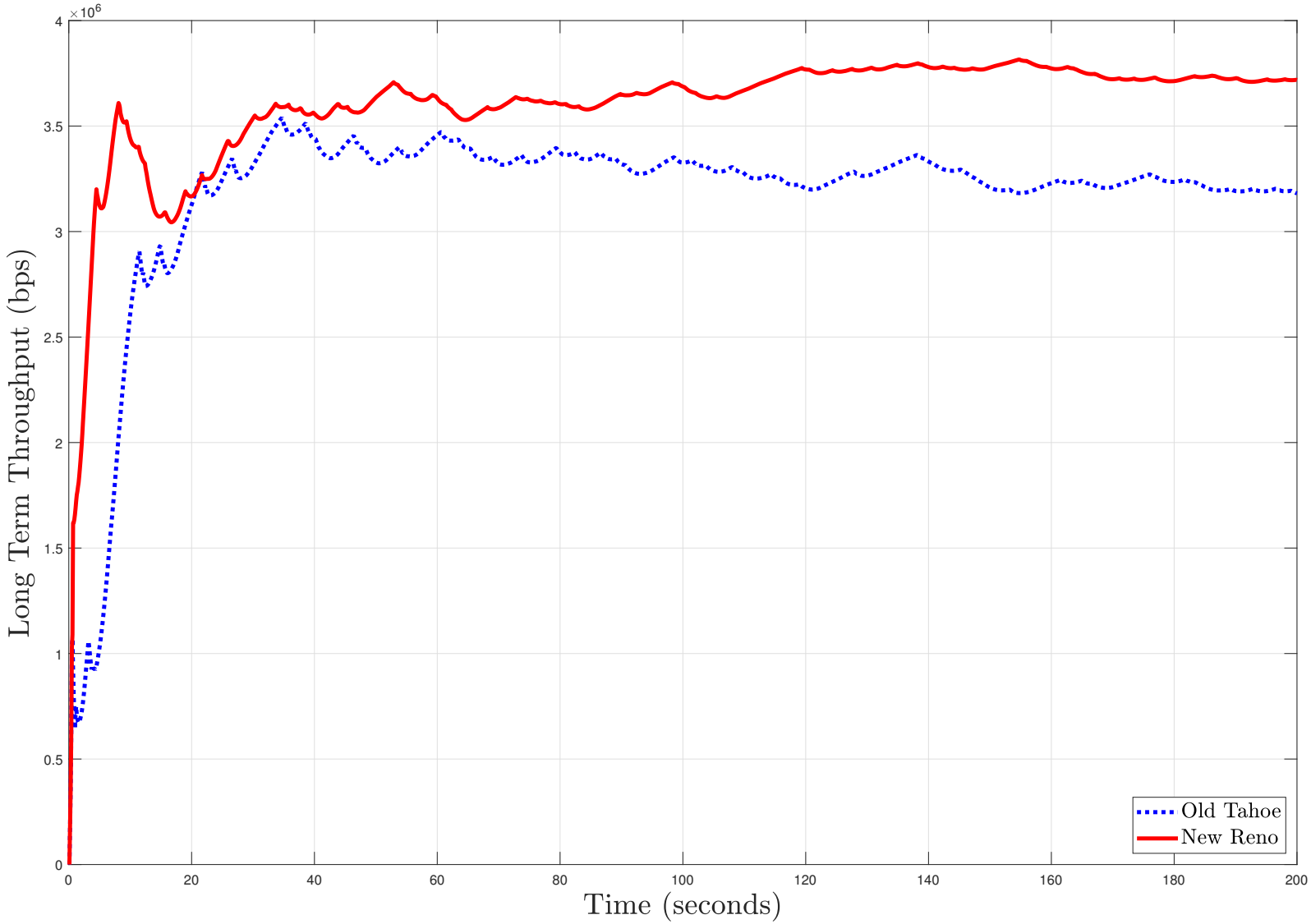


TCP Congestion Control: New Reno

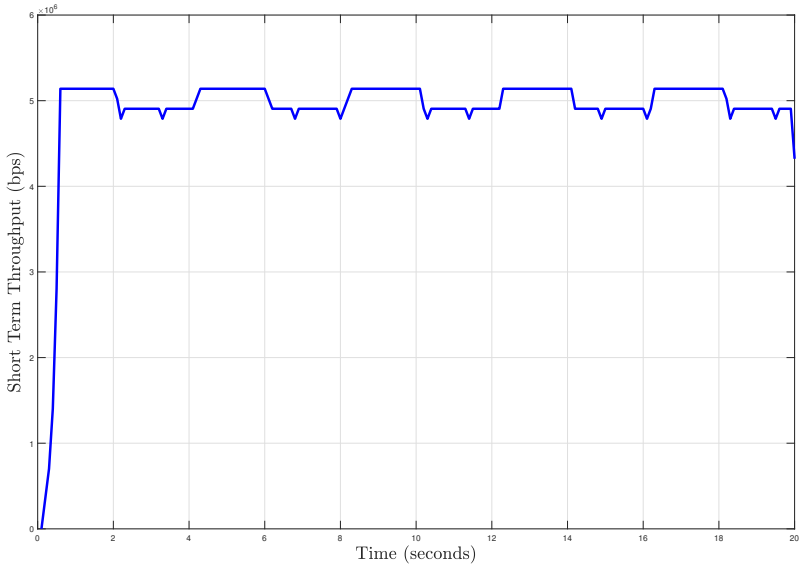


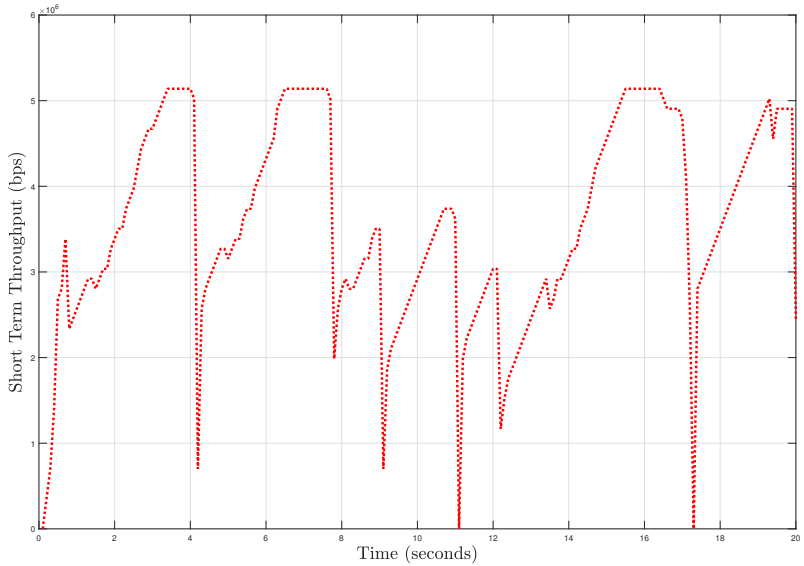
source: Computer Networking: A top-down approach, Kurose and Ross

TCP Congestion Control: Performance Comparison



Throughput Performance with TCP Reno





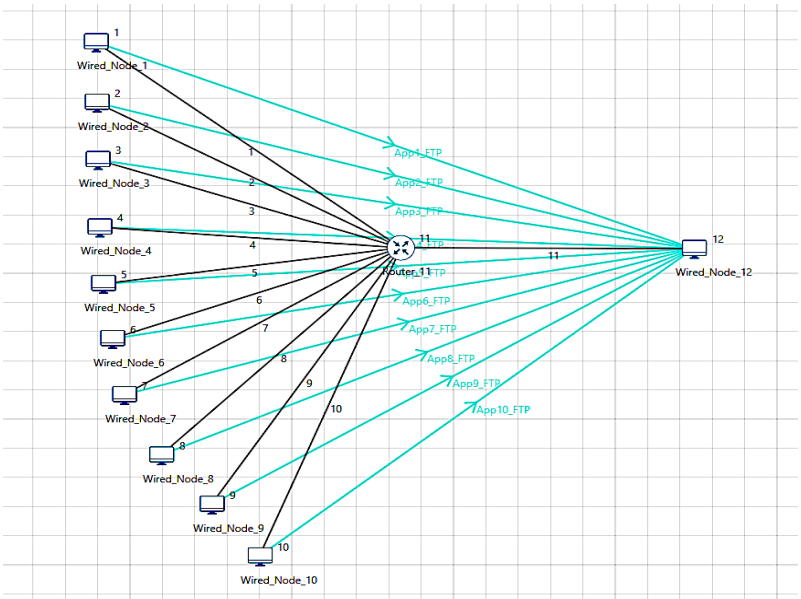
- Approximate analysis of throughput
  - loss-less links: throughput $= \frac{\text{cwnd}}{\text{RTT}}$
  - lossy link: throughput $= \frac{1.22\ \text{MSS}}{\text{RTT}\sqrt{p}}$

TCP Congestion Control: Inference

- TCP congestion control
  - end-to-end congestion control
  - responds to delays and duplicate acknowledgements
  - affects long-term average throughput and fairness
  - recent and popular versions include Cubic, Compound and HTCP
- Simulator
  - congestion window evolution observed with PCAP file and packet trace

Multiplexing and Fairness

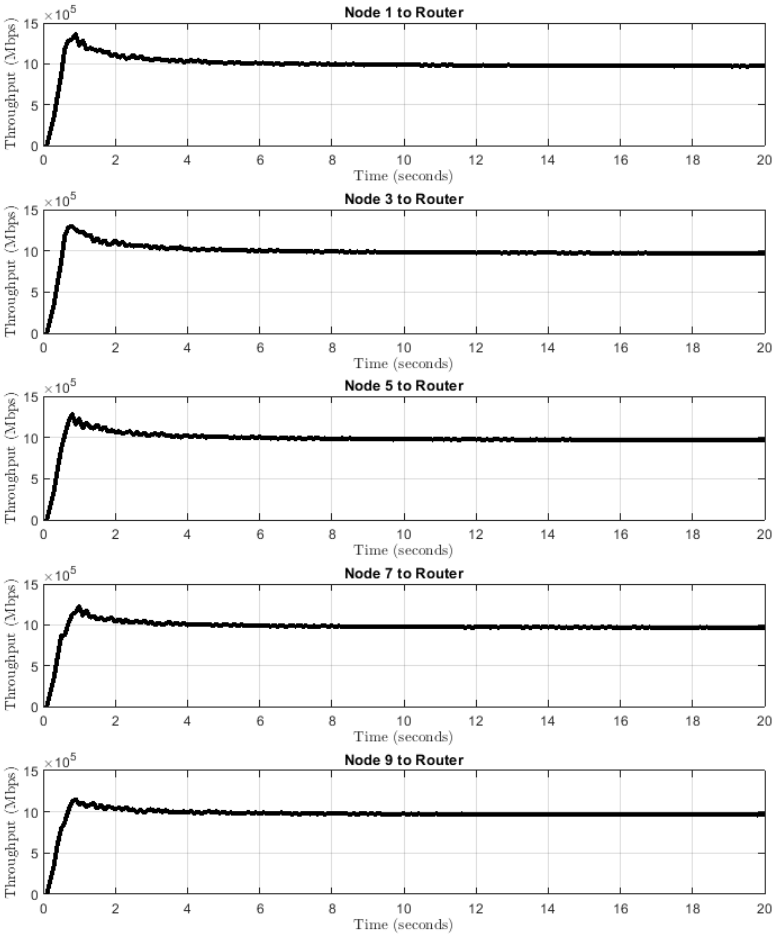
- Objective
  - study network resource sharing with multiple TCP flows
  - focus on throughput and fairness
- Network setup
  - ten clients connect to a server over loss-less links
  - FTP application seeks to transfer a very large file using TCP

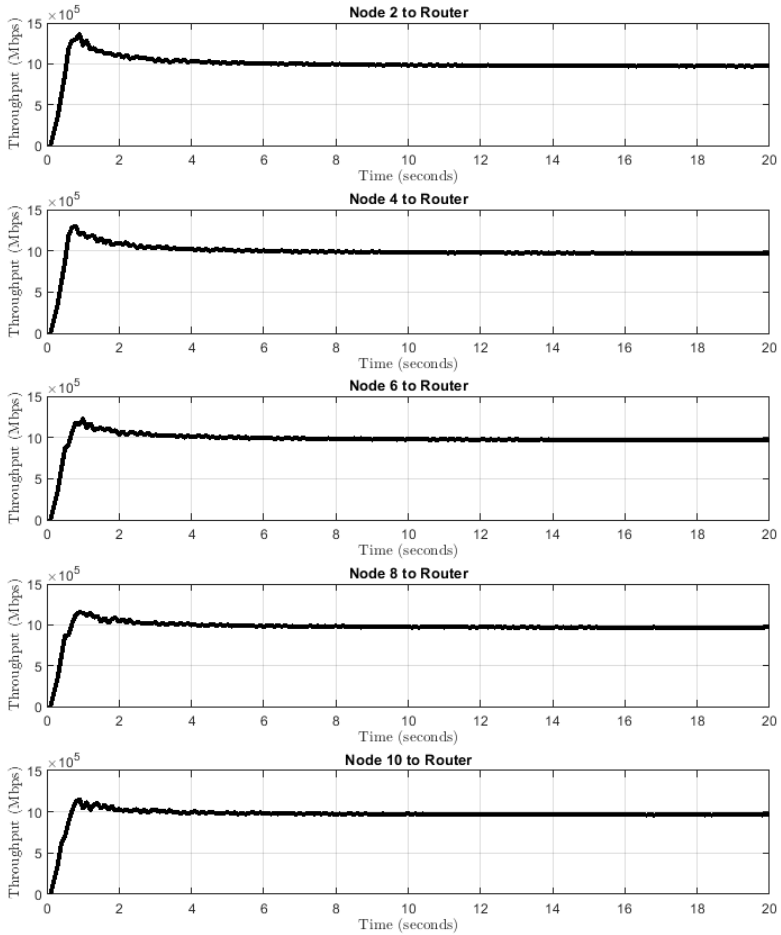


Multiplexing and Fairness: Experiment Configuration

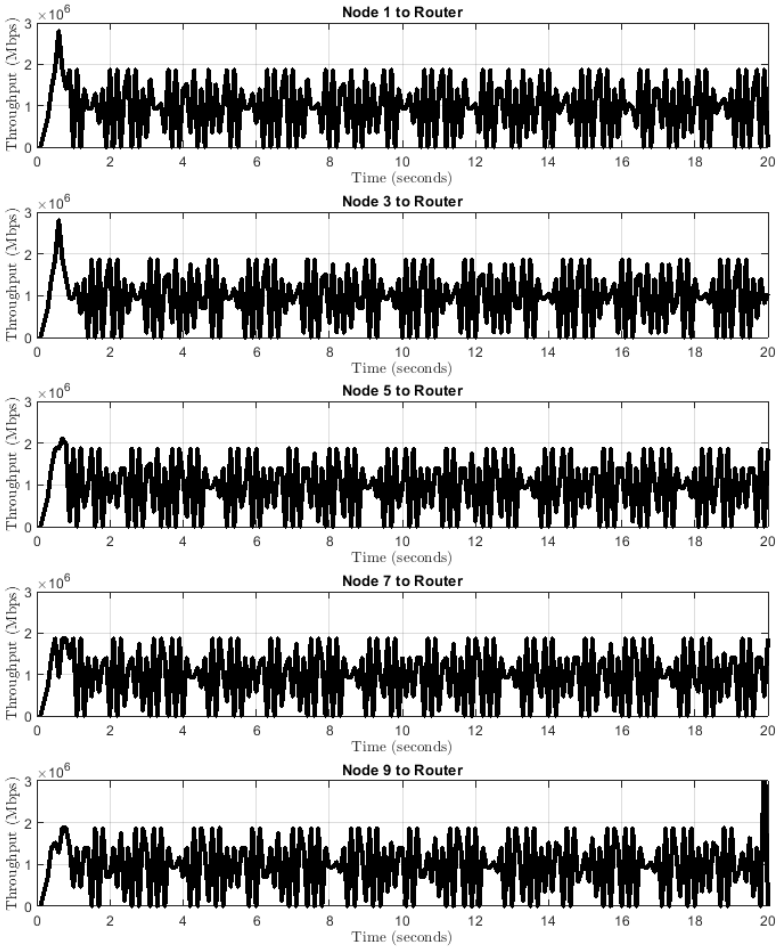
Link Parameters	
Wired link speed	10 Mbps
Wired link BER	0
Wired link propagation delay	25 msecs
Transport Layer Parameters	
Transport Protocol	TCP
Congestion Control	New_Reno
MSS	1460
Application Parameters	
Application	FTP
File Size	10^8 bytes
Miscellaneous	
Simulation Time	20 secs
Wireshark	Enabled On

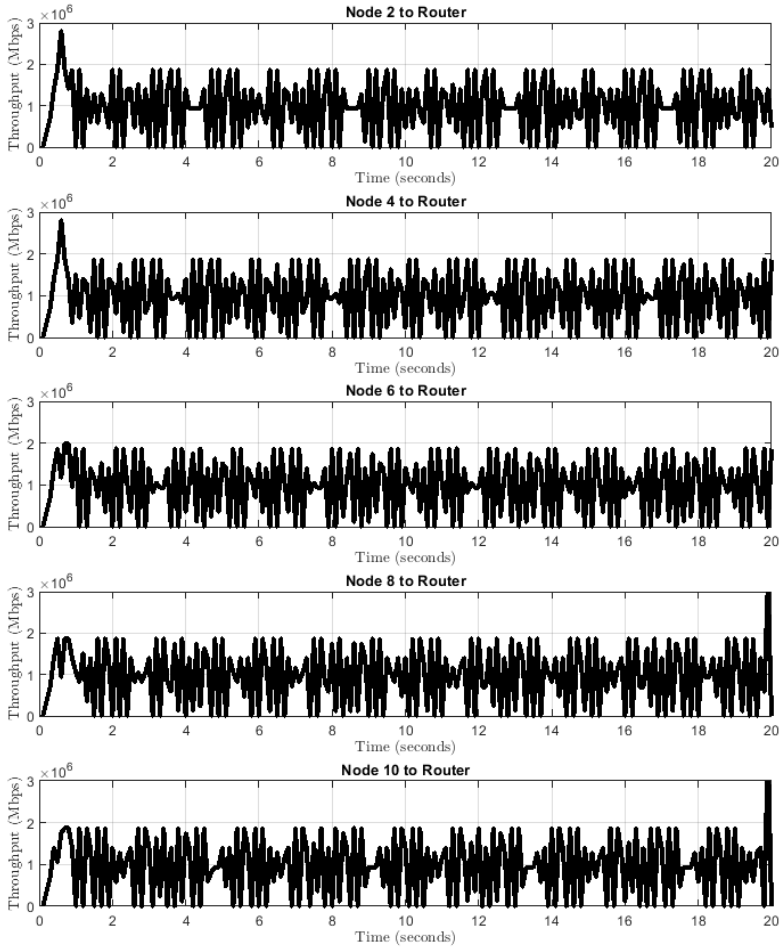
Multiplexing and Fairness: Average Throughput





Multiplexing and Fairness: Short-term Average Throughput





Multiplexing and Fairness: Inference

- Network multiplexing with TCP
  - network resources are shared equally in long-term
  - network sharing is unfair in short-term
  - performance critically depends on link capacity, buffer, algorithm, congestion, etc
  - queue management at routers can help improve performance
- Simulator
  - performance observed with packet trace

Summary

- Transmission control protocol
  - provides end-to-end connection-oriented communication service
  - key functions are reliability, congestion control and multiplexing
  - dynamically adapts to network congestion
  - recent versions include Cubic, Compound, H-TCP
- Recent advances in transport layer protocols
  - QUIC: quick UDP Internet connections
  - computer generated congestion control algorithms

Suggested Exercises

- Bandwidth-delay product (BDP)
  - performance of TCP depends on BDP
  - buffer size at the bottleneck routers must be appropriately set
- Multiple flows
  - TCP vs TCP: performance depends on RTT, rwnd, etc
  - TCP vs UDP: UDP does not support congestion control
- TCP and Wireless
  - TCP perceives packet loss as congestion
  - flows may interact at buffer as well as the wireless channel
- TCP in a network
  - resource sharing in a network
  - performance with ECN, RED, WFQ

References

- NetSim
  - homepage - https://www.tetcos.com/
  - videos - https://www.youtube.com/user/Tetcos/videos
  - facebook - https://www.facebook.com/tetcosnetsim/
- TCP specification
  - RFC 793 (protocol), RFC 5681 (congestion control)
- Articles
  - A protocol for packet network intercommunication, V Cerf and R Kahn, 1974
  - Congestion avoidance and control, V Jacobson, 1988
- Text
  - Computer networking: A top-down approach, J Kurose and K Ross
  - TCP/IP Illustrated, Volumes I, II and III, W R Stevens, et al

Questions!


obrigado
Dank U
Merci
mahalo
Köszi
спасибо
Grazie
Thank you
mauruuru
Takk
Gracias
Dziękuję
Děkuju
danke
Kiitos