# Computer Networks
# COL 334/672

Flow control, Congestion control

*Slides adapted from KR*

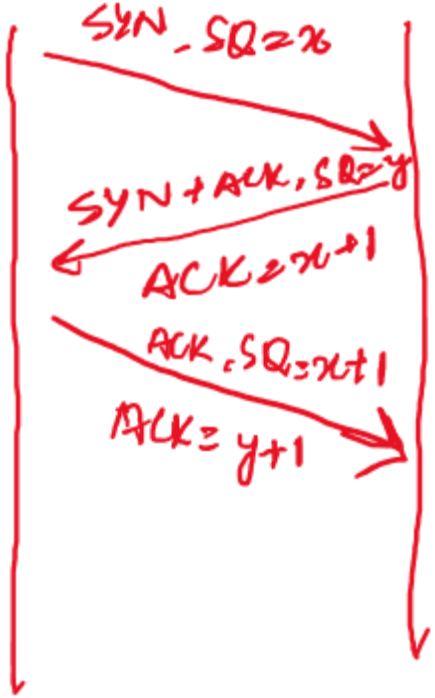
Sem 1, 2025-26

# Recap: TCP Connection Establishment

(1) Both sides need to agree on connection

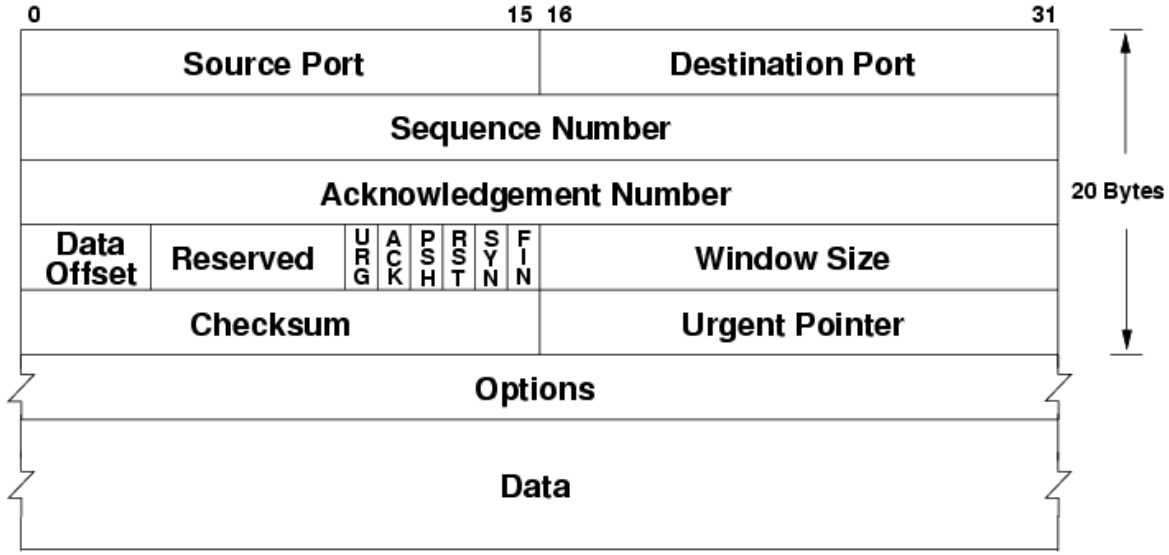
(2). Agree on Initial set of sequence #s

(1). 1 RTT for connection establishment




6 Flags

- SYN
- ACK
- FIN
- RST
- URG
- PUSH



# Why use random sequence numbers?


[stale SYN / ACKs / DATA] are an issue

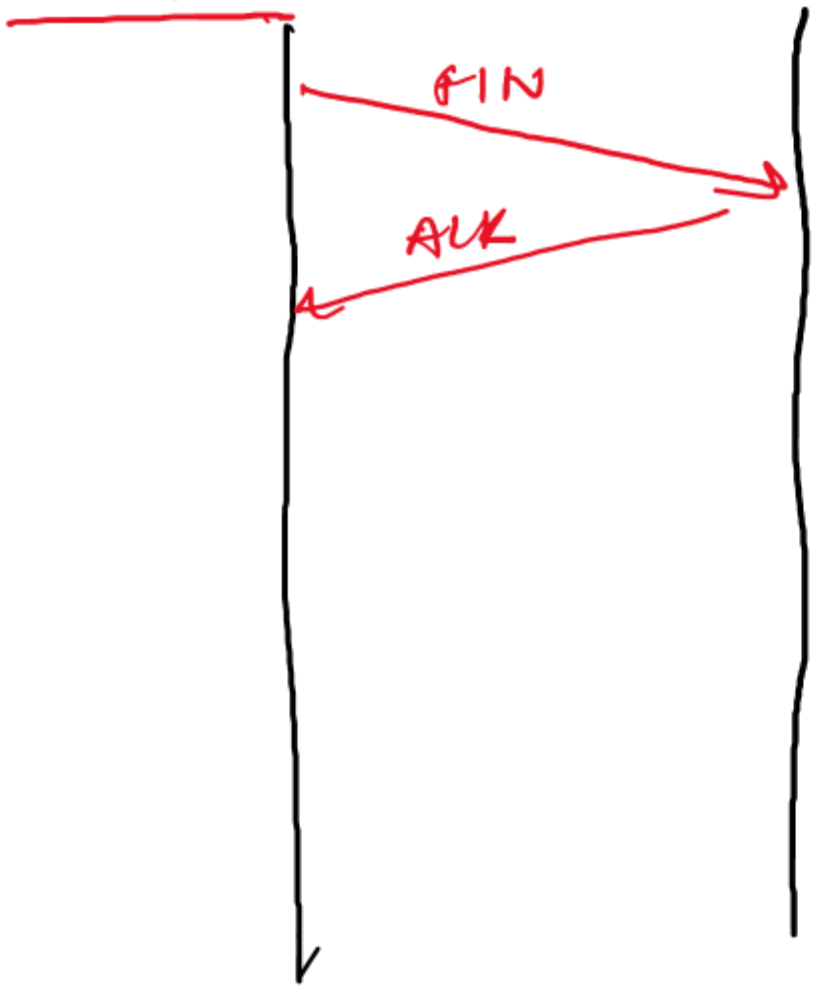
→ Using Random sequence #s can help with that

→ half-open connections could still be an issue

Data will be sent to upper layers

# How to close a TCP connection?

DESIGN-1




Limitations

① Synchronous close is not possible (2-Army problem)

② What if other side has data to send

DESIGN-2 : Each side closes connection independently


4-way handshake

# TCP Functions

- Connection establishment
- Reliability
- **Flow control**
- Congestion control

# Flow Control

Rate of arrival of data ≥ Rate of retrieval by the appli

What happens?

Buffer overflow

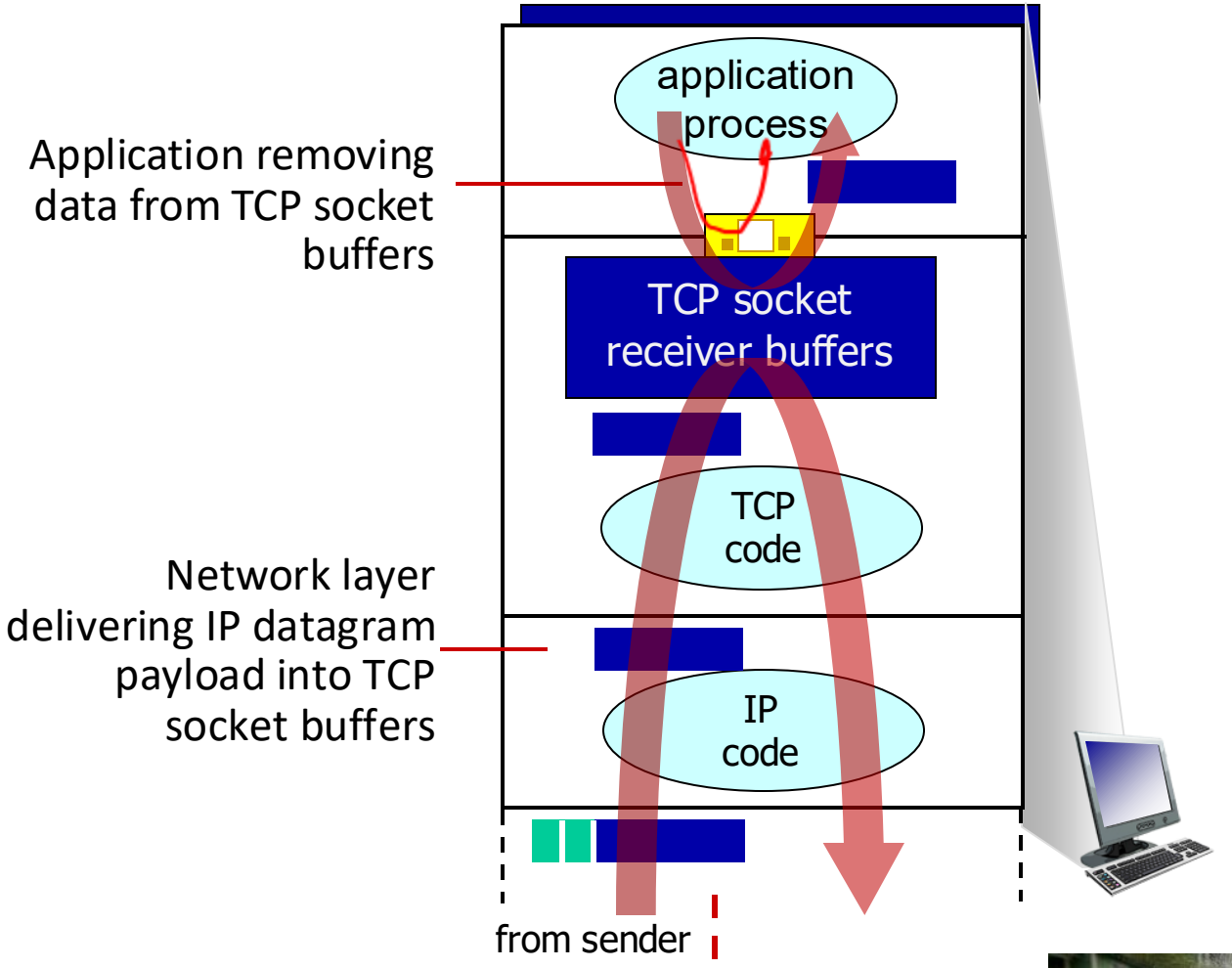
↳ wasted B/W

100KB buffer

5KB

**flow control**

↳ receiver controls sender, so sender won't overflow receiver's buffer by transmitting too much, too fast




receiver protocol stack

# TCP Flow Control Mechanism

rwnd

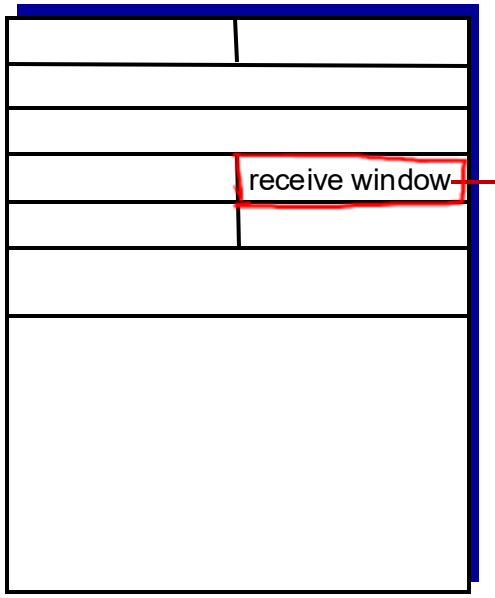
sliding window ≤ rwnd

(Reliability)

Rate control at the sender

goal: Not overwhelm the receiver

16-bit header.




flow control: # bytes receiver willing to accept

# TCP Functions

- Connection establishment
- Reliability
- Flow control
- **Congestion control**

# Congestion Control

→ control sending rate so NOT overwhelm the n/w router

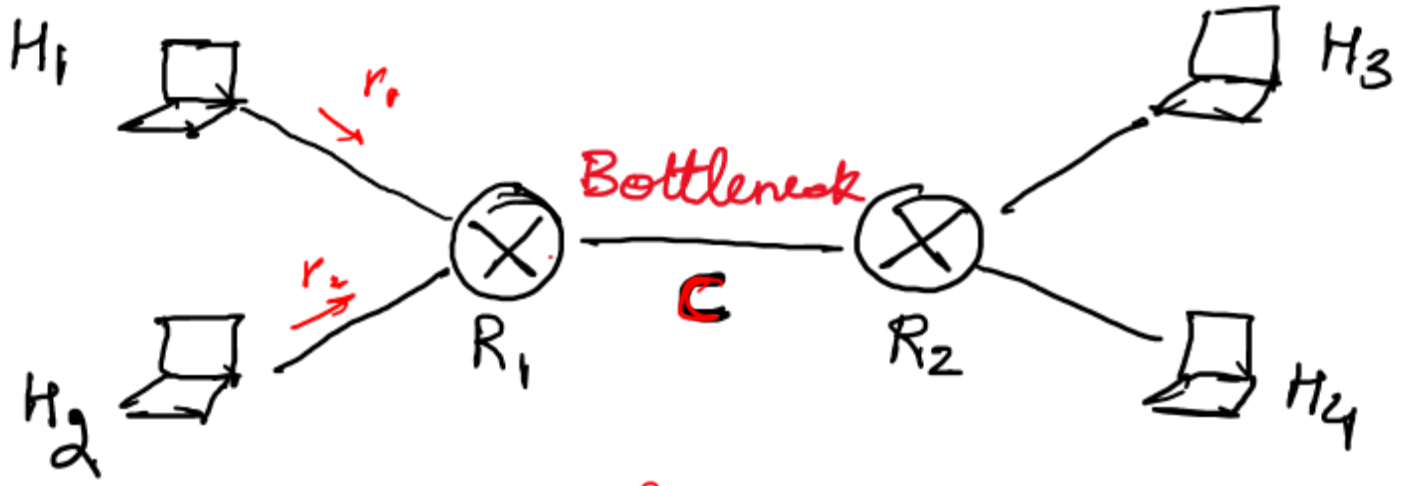
## Congestion:

- informally: "too many sources sending too much data too fast for *network* to handle"
- different from flow control!
- a top-10 problem!

congestion control:
too many senders,
sending too fast

# What/Why Congestion Control?

Dumbbell Topology




$r_1 + r_2 \geq C$ : congestion happen

$r_1 + r_2 < C$ : such that

Congestion control

Why is it important

① Wasted Bandwidth

② Queuing delay will increase

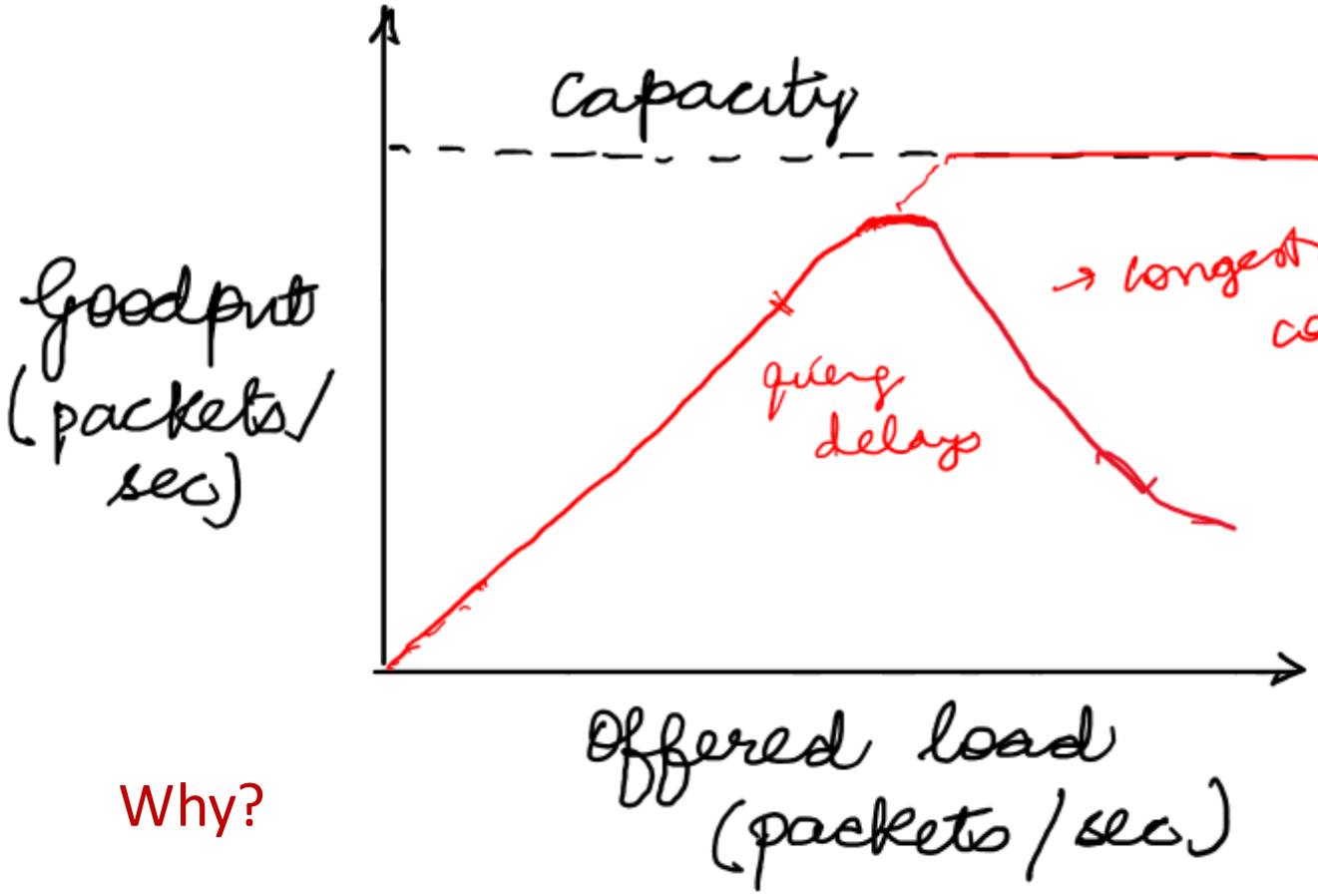
③ Fairness is compromised

④ Retransmission

# History: The Case of Congestion Collapse

Sliding window:

- Early TCP protocol in 1980s used fixed size window
  - The focus was mainly on providing reliability
- As network load grew, goodput reduced



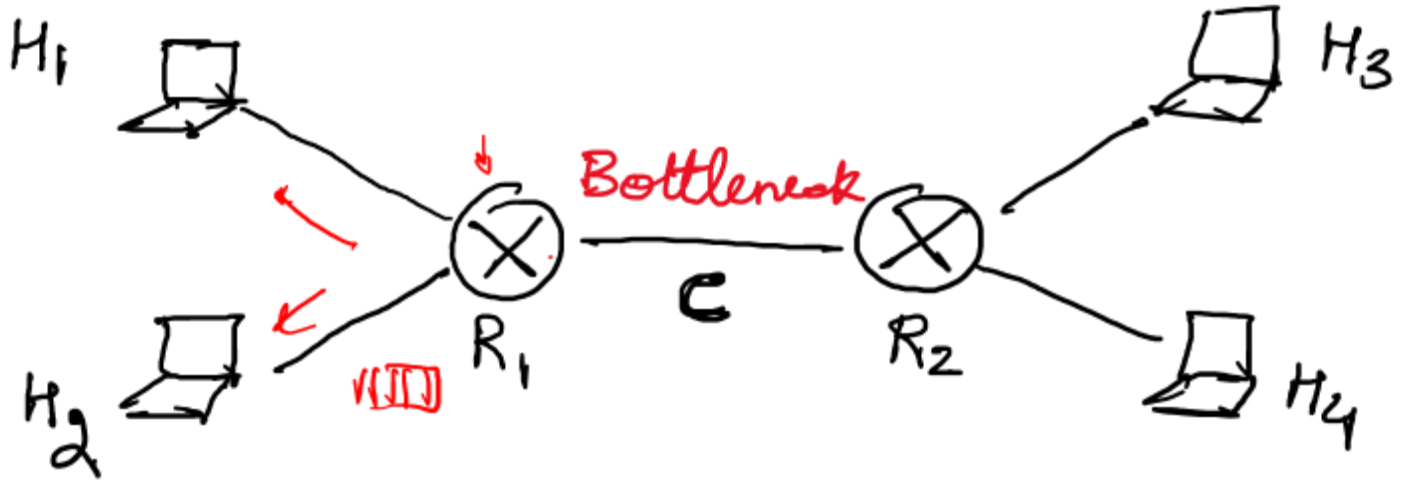

Why?

"costs" of congestion:

- more work (retransmission) for given receiver throughput
- unneeded retransmissions: link carries multiple copies of a packet
  - decreasing maximum achievable goodput

**Need mechanisms for congestion control**

# Congestion Control Algorithm (CCA) Approaches




WHOSE PROBLEMS

Both end hosts and routers

N/w assisted

1. Routers participate in congestion control along w/ end host

E2E

1. Only end-host perform congestion control