CSE 461
Network-Side Congestion Control

Chunjong Park

End-to-End Congestion Control

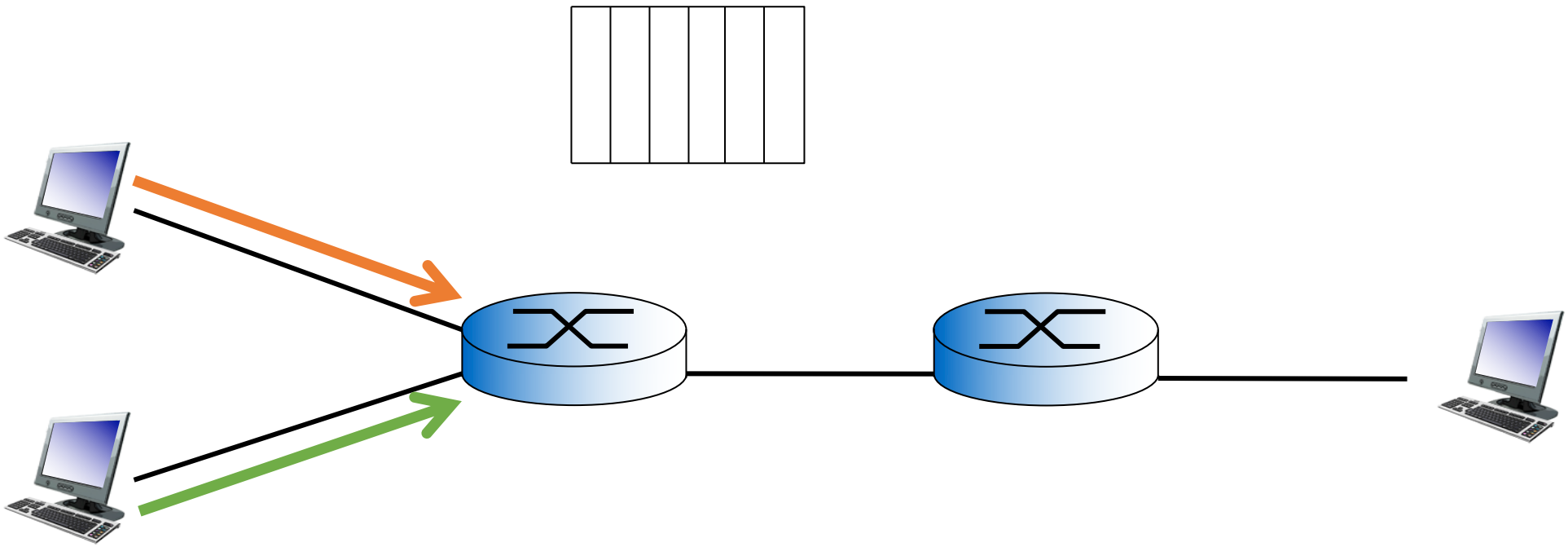
- TCP
  - Treats the network as a “black-box”
  - Relies on packet loss as the indication of congestion
- Congestion detection
  - Three duplicate ACKs
  - Retransmission timeout
- Congestion avoidance
  - Multiplicative decrease

- ***Reaction after congestion has already happened!***

Limitations of End-to-End Congestion Control

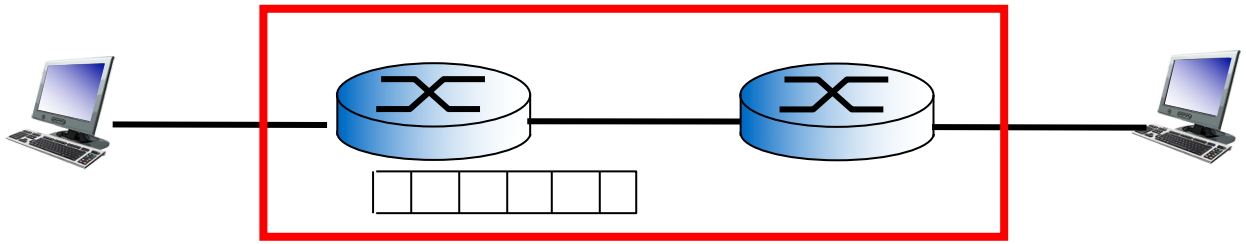
- Not helpful for delay-sensitive flows
  - The increased latency of the packet is caused by the need to ***retransmit the packet after a loss***
- Undesirable global synchronization
  - All flows reduce the sending rate ***simultaneously*** → channel is under-utilized
  - All flows start retransmission/***increasing the sending rate*** in a similar fashion → congestion occurs again
- Lockout
  - The shared resource is unfairly consumed exclusively by a ***small number of flows***
  - The remaining flows are denied access to the resource

Limitations of End-to-End Congestion Control



Network-Side Congestion Control

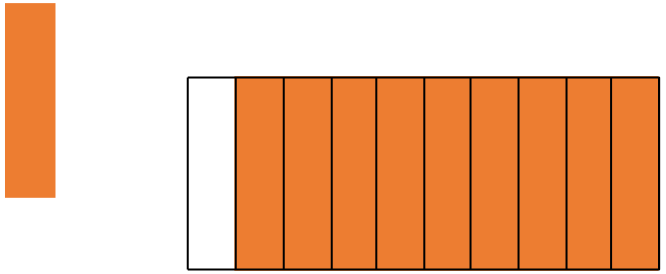
- Network-side/network-assisted
  - Switches and routers

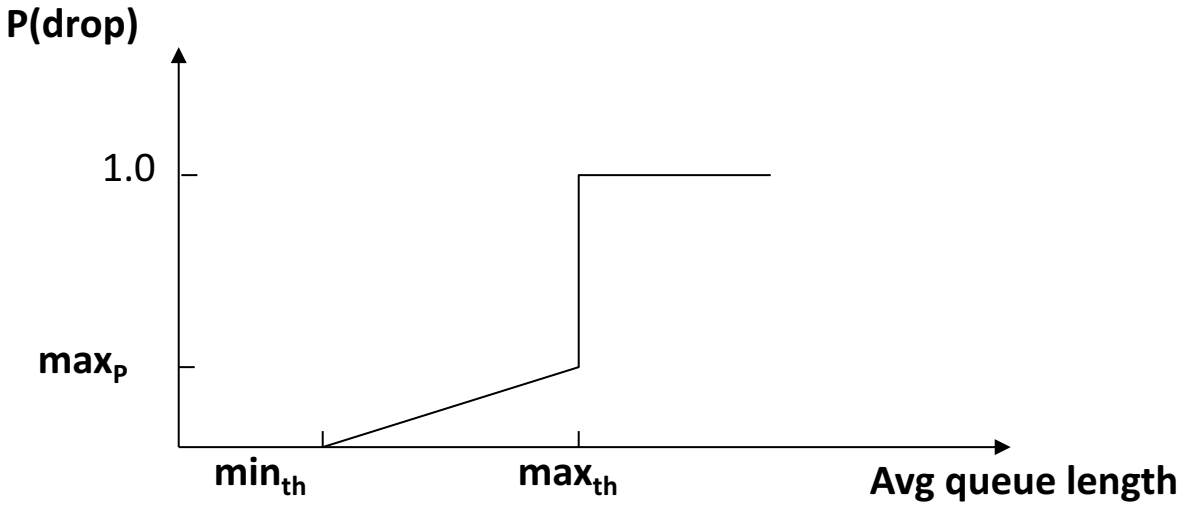


- Routers detect congestion before the buffer is full
  - Beginning of congestion indicated by the queue size
- The router provides an indication to the hosts before **congestion loss** really happens
  - Drop packets → Radom Early Detection (RED)
  - Mark packets → Explicit Congestion Notification (ECN)

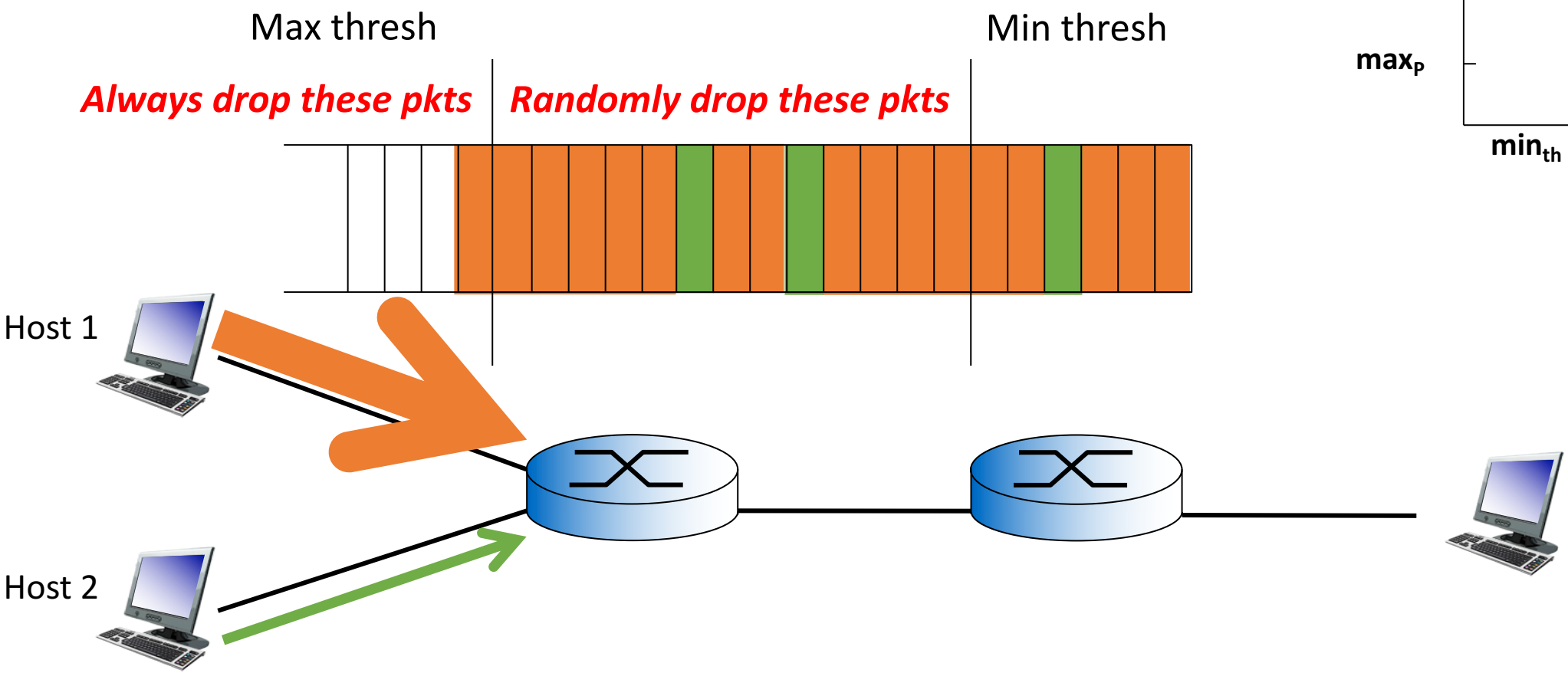
Random Early Detection

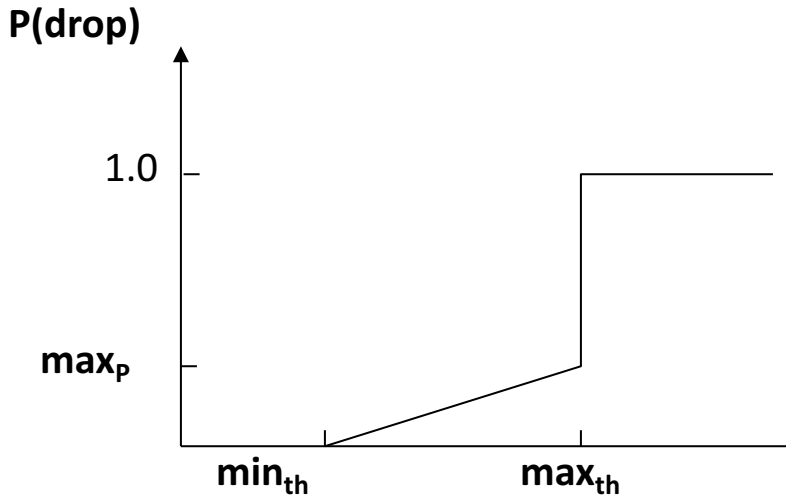
- When router's buffer is filling, drop TCP packets at random
- Drop at random, depending on queue size
  - If queue empty, accept packet always
  - If queue full, always drop
  - As queue approaches full, increase likelihood of packet drop
  - Example: 1 queue slot left, 10 packets in a buffer, 90% chance of drop





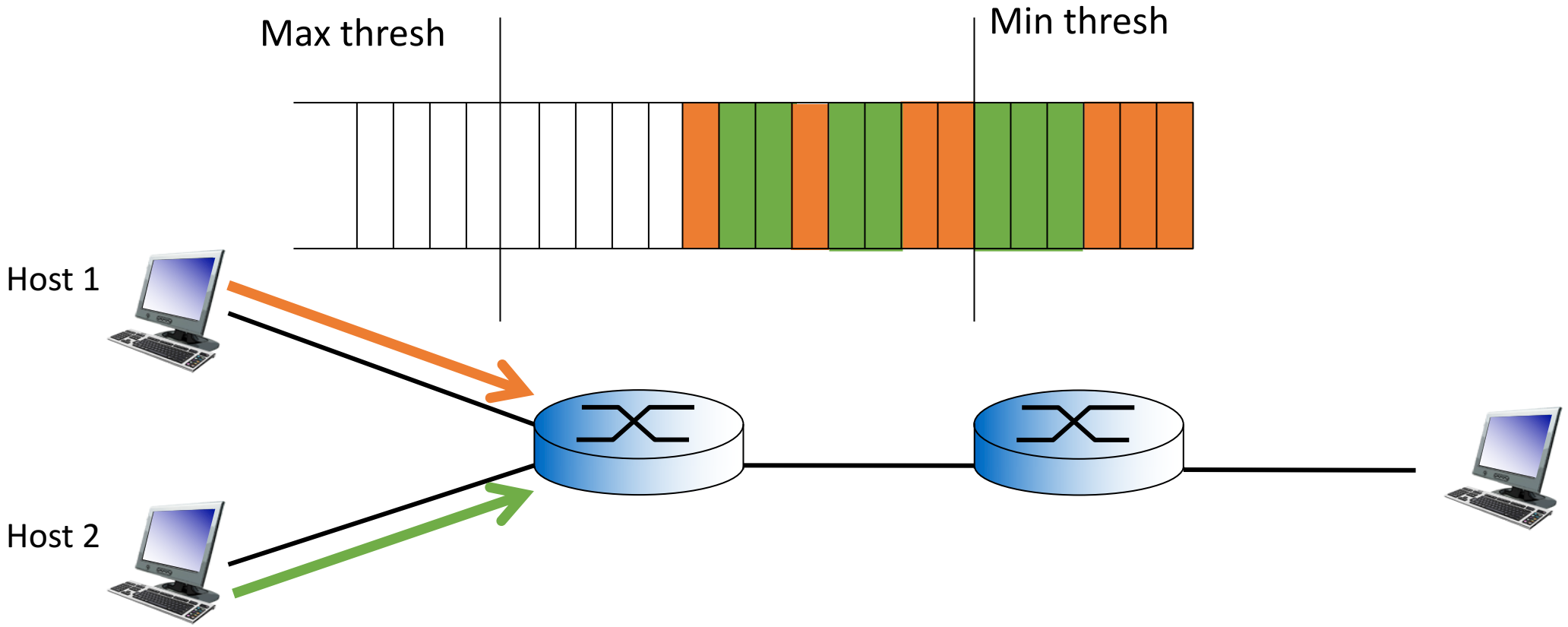
Random Early Detection





- When you pick a packet at random to drop, which flow is it most likely to belong to?

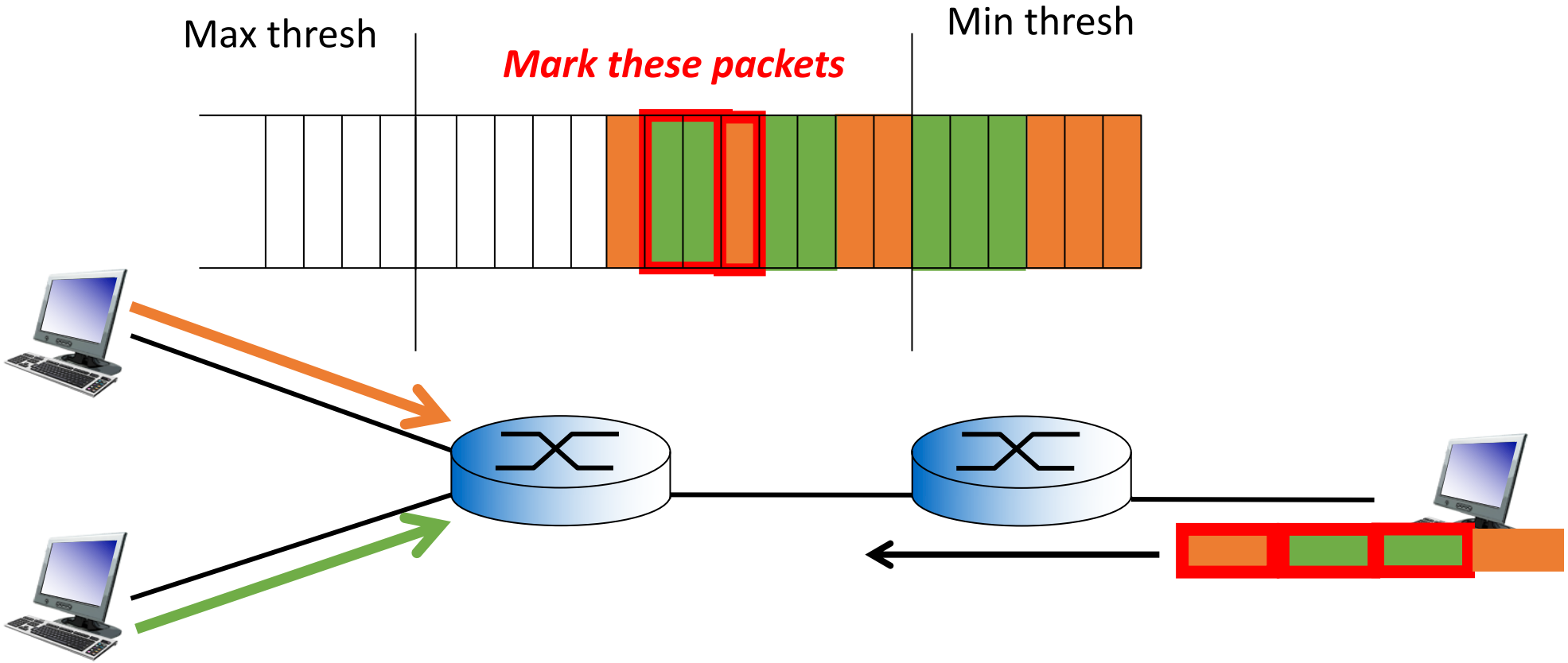
Random Early Detection



- Host 1 will decrease cwnd size and eventually will fairly share the link

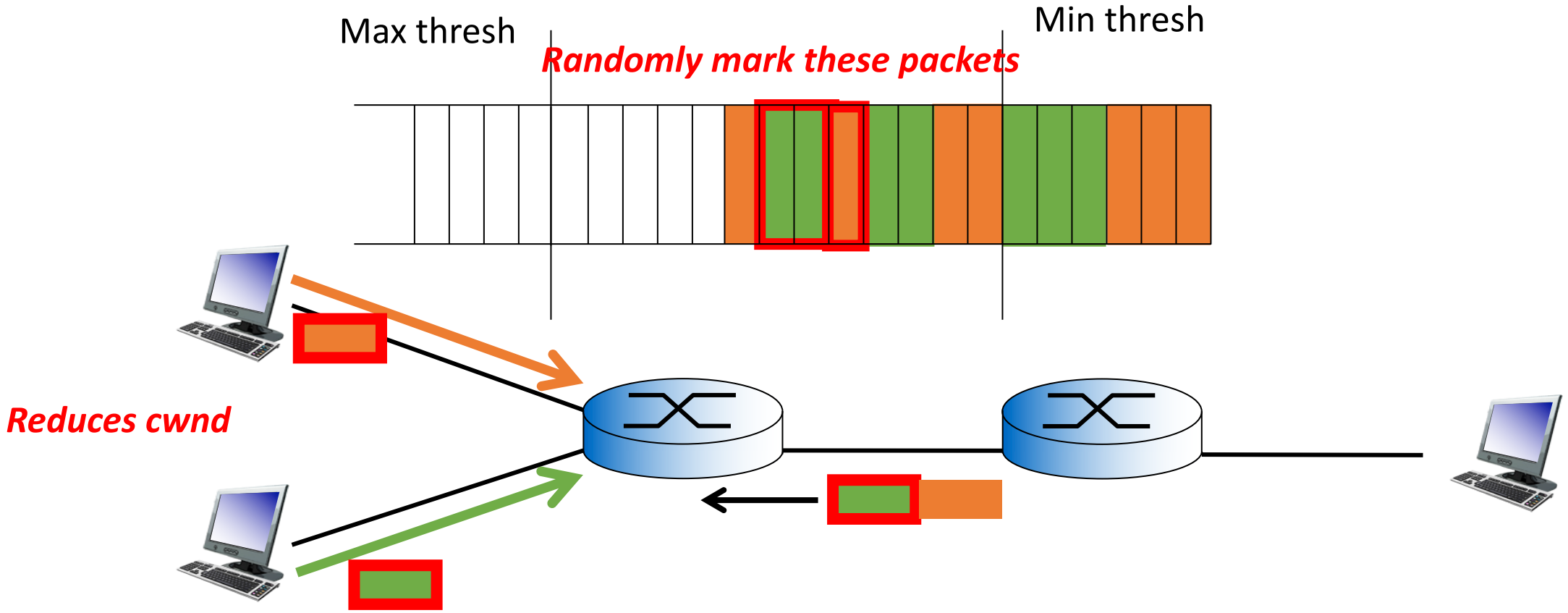
Explicit Congestion Notification (ECN)

- **Idea**: to send congestion feedback to sender when switch/router buffer is about to be full

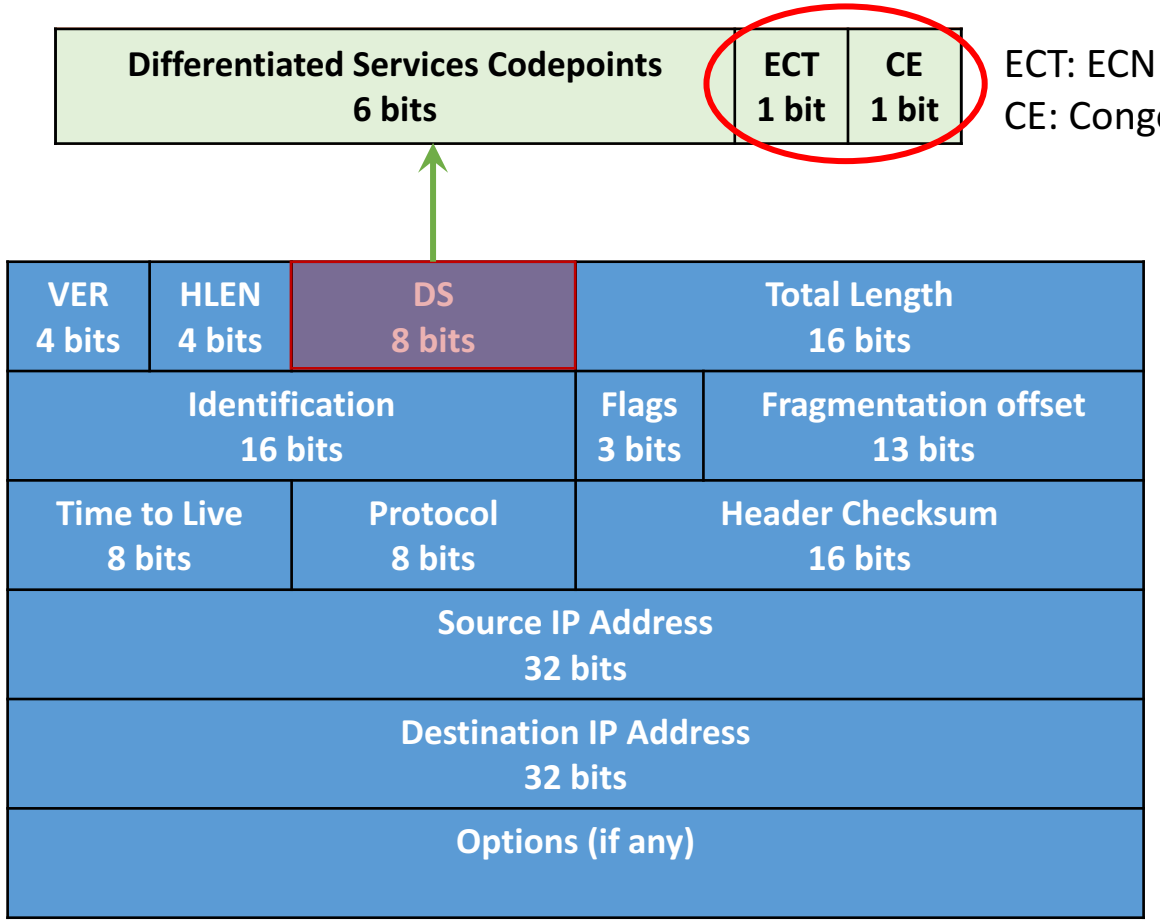


Explicit Congestion Notification (ECN)

- **Idea**: to send congestion feedback to sender when switch/router buffer is about to be full



Where do we mark? (IP header)

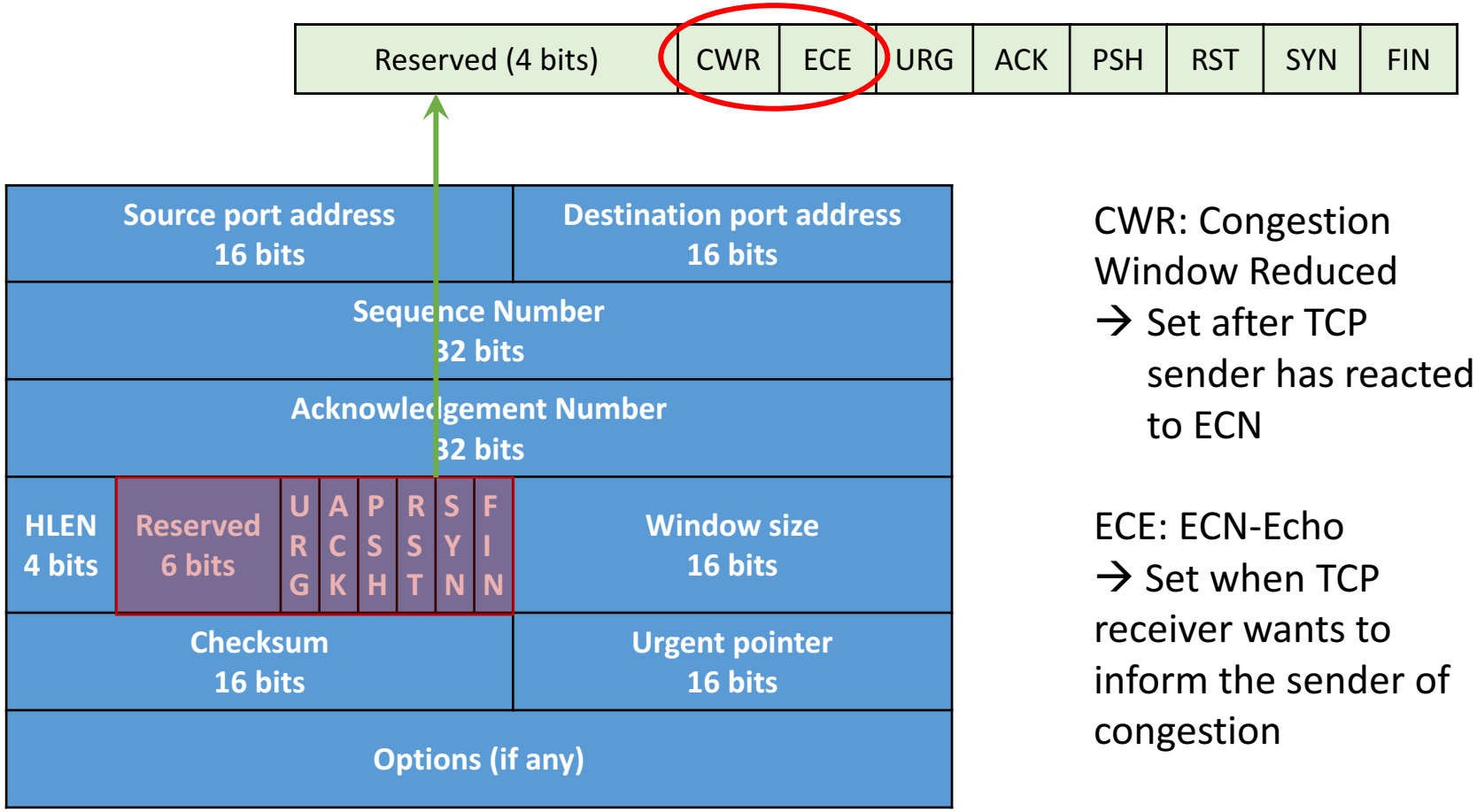


ECT: ECN Capable Transport
CE: Congestion Experienced

ECT	CE	
0	0	Not-ECT
0	1	ECT (1)
1	0	ECT (0)
1	1	CE

Source: ECN lecture by Qi (Gill) Wang (gillwang@udel.edu)

Where do we mark? (TCP header)

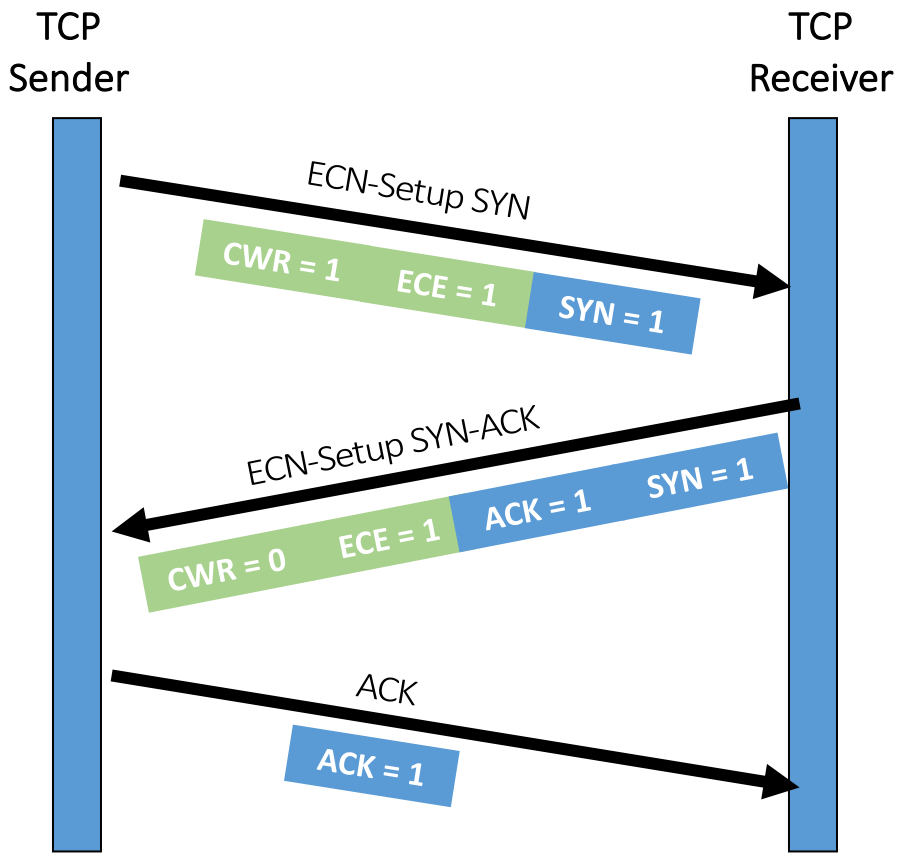


CWR: Congestion Window Reduced
→ Set after TCP sender has reacted to ECN

ECE: ECN-Echo
→ Set when TCP receiver wants to inform the sender of congestion

Source: ECN lecture by Qi (Gill) Wang (gillwang@udel.edu)

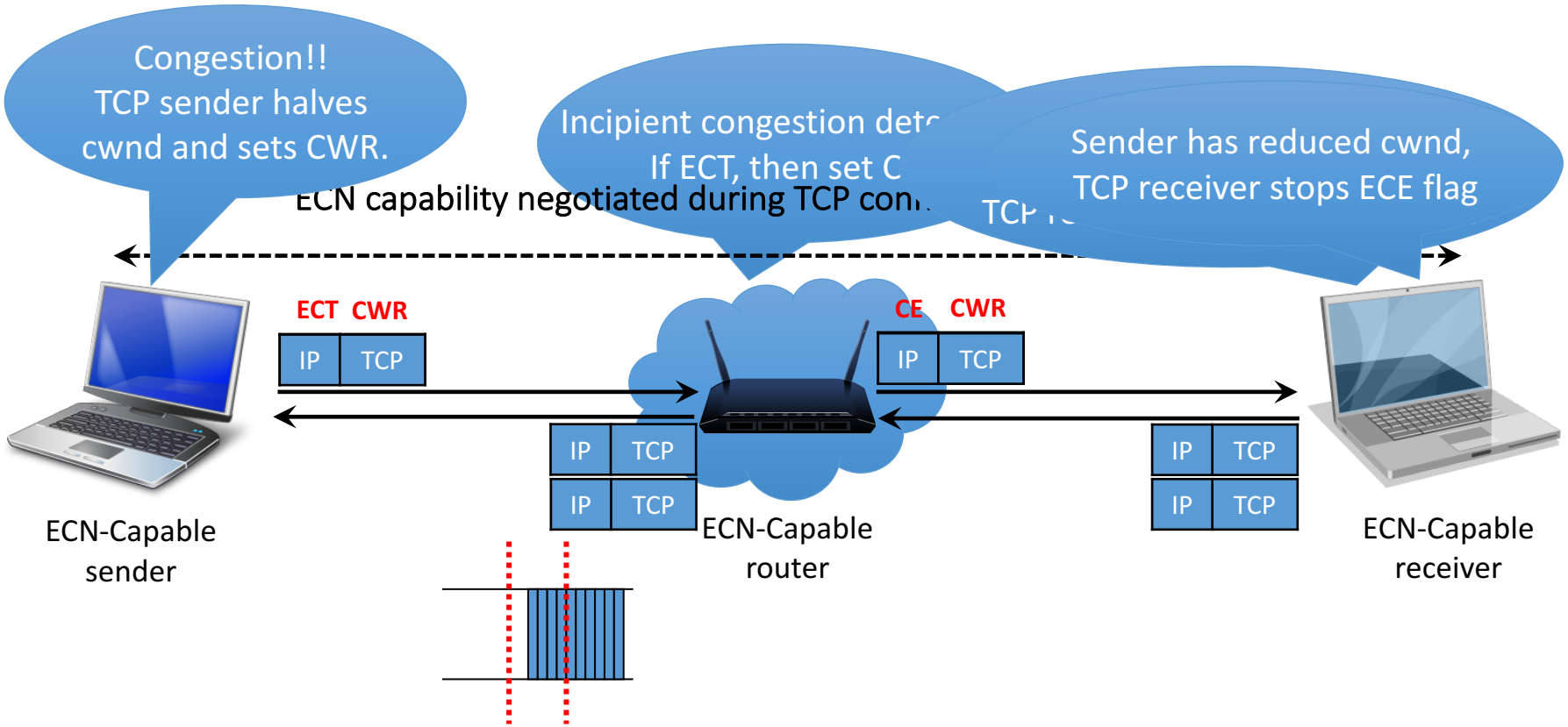
ECN Negotiation in TCP



Source: ECN lecture by Qi (Gill) Wang (gillwang@udel.edu)

ECN Congestion Control

ECT	**ECN Capable Transport**
CE	**Congestion Experienced**
CWR	**Congestion Window Reduced**
ECE	**ECN-Echo**



Source: ECN lecture by Qi (Gill) Wang (gillwang@udel.edu)

ECN Advantages

- Feedback to the TCP sender before the router drops packets
- Less retransmission
- Improve throughput and goodput of the whole internetwork

Source: ECN lecture by Qi (Gill) Wang (gillwang@udel.edu)

Summary

- Limitations of end-to-end congestion control
  - Increased latency
  - Undesirable global synchronization
  - Lockout
- Network-side congestion control
  - Switches and routers can detect beginning of network congestion
  - Indicates congestion before loss/timeout
  - Random Early Detection
  - Explicit Congestion Notification

Backup

TCP Slow Start Graph