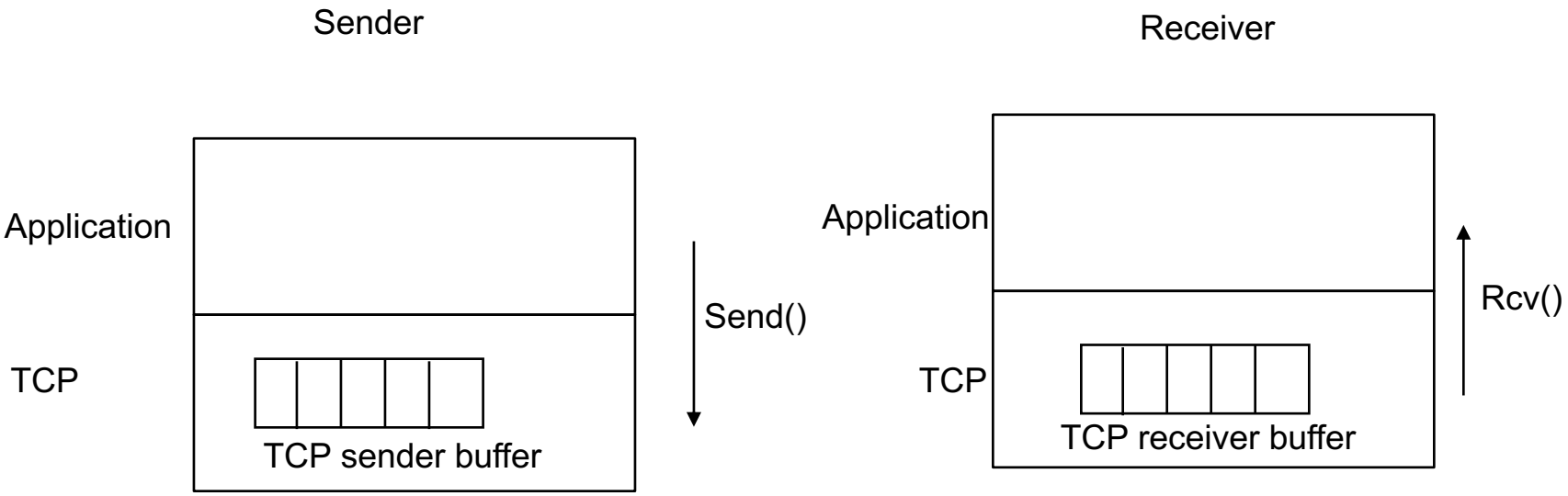
Transport layer: Flow Control

ECE 50863 – Computer Network Systems

Flow control

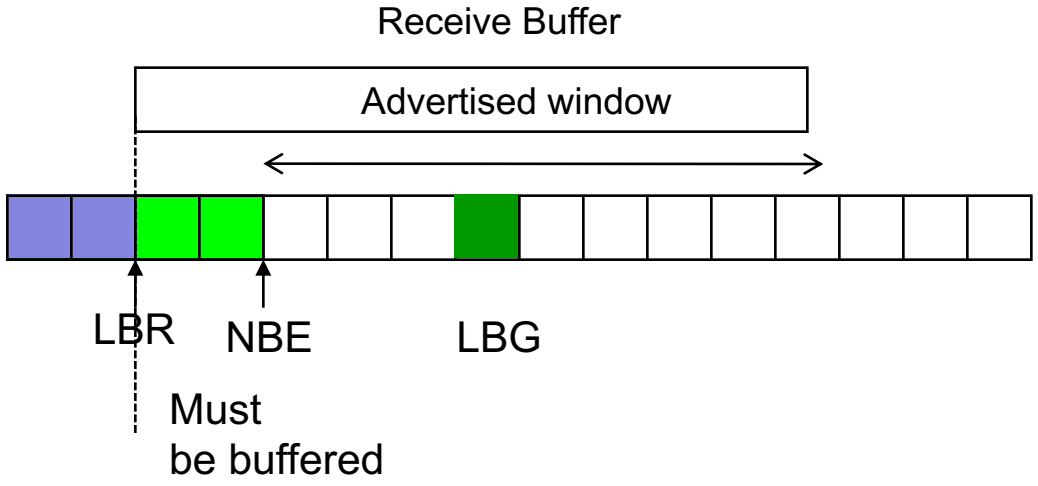
- Consider that the sender sends too fast for the receiver
- Flow control: a mechanism by which receiver can throttle the sender.
- Distinct from congestion control
  - Sender is too fast for an intermediate network link.
  - Slows down to avoid too much traffic on that link.

Flow Control

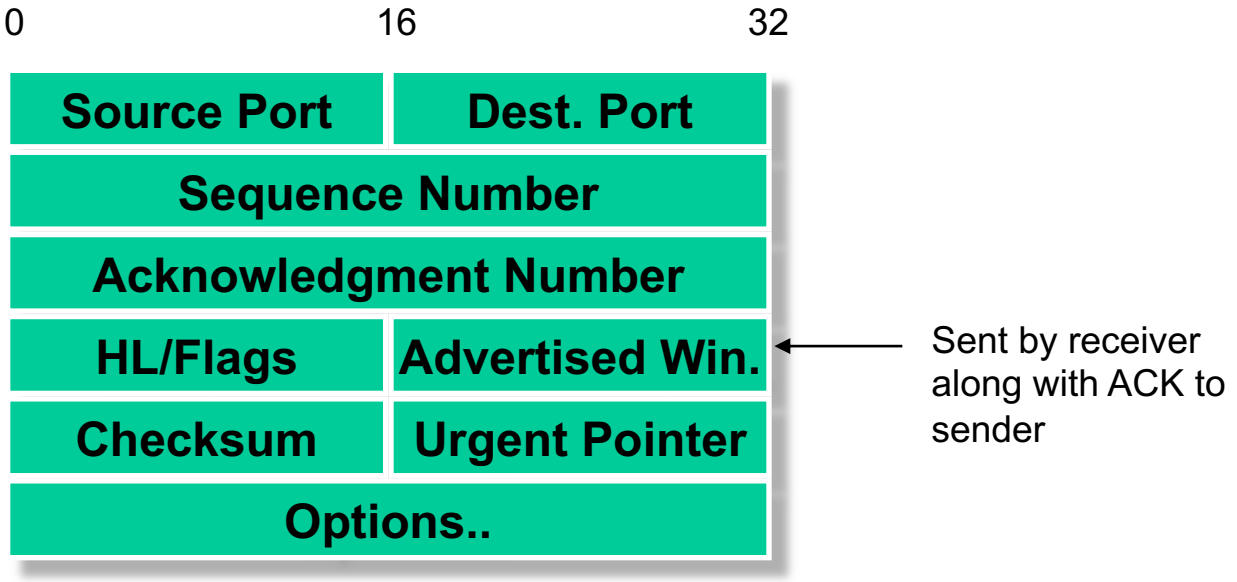


Flow Control

- Receiver throttles the sender by advertising a window no larger than the amount of data it can buffer
- Advertised Window depends on
  - Size of receive buffer
  - How fast receiver process can read the bytes from the buffer
- TCP sending window impacted by advertised window

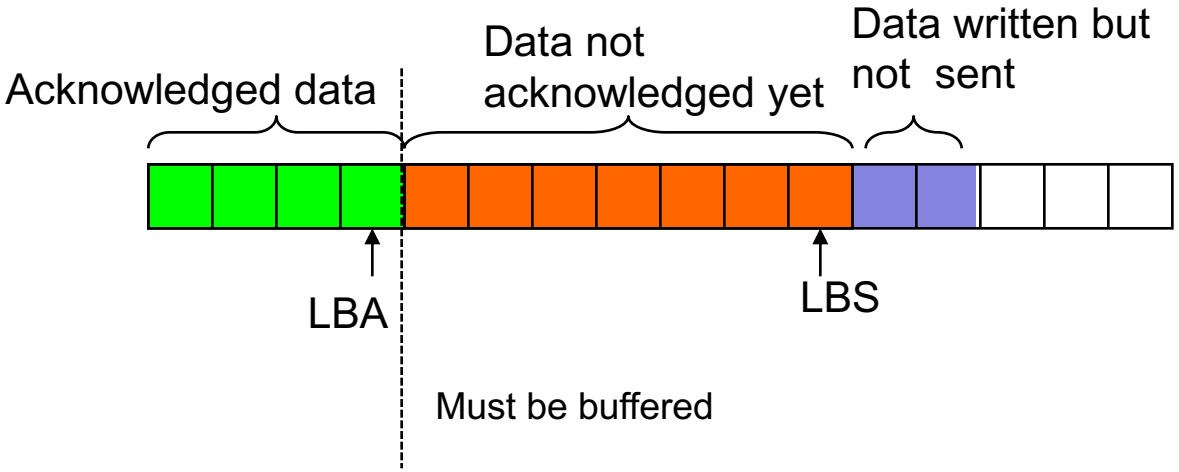


TCP Header

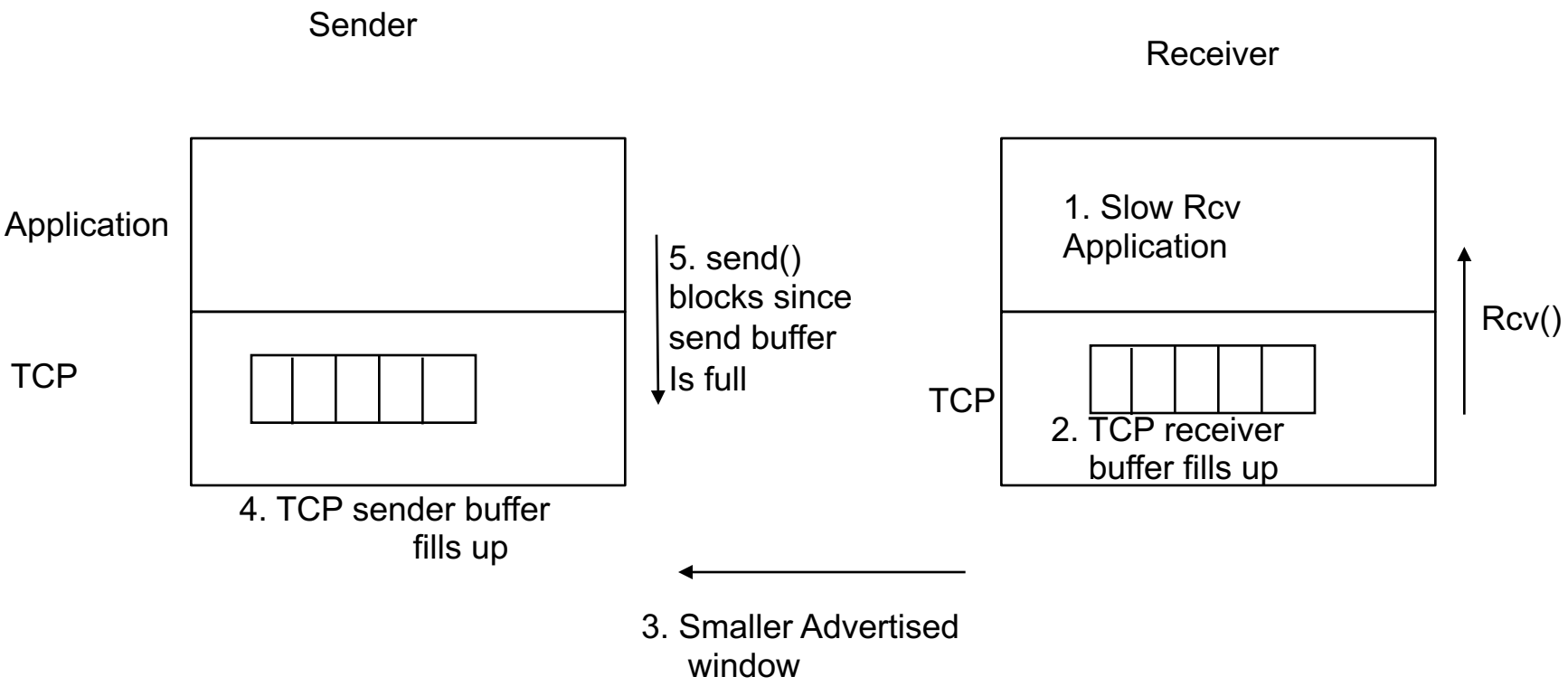

0
16
32
Source Port
Dest. Port
Sequence Number
Acknowledgment Number
HL/Flags
Advertised Win.
Checksum
Urgent Pointer
Options..
Sent by receiver along with ACK to sender

Flow Control (sender side)

- Receiver throttles the sender by advertising a window no larger than the amount of data it can buffer
- Effective window at sender (limits how much it can send)
  - Advertised window – (outstanding unacknowledged data)
- Sending process can write to send-side TCP buffer only if space in sender side buffer – otherwise will block



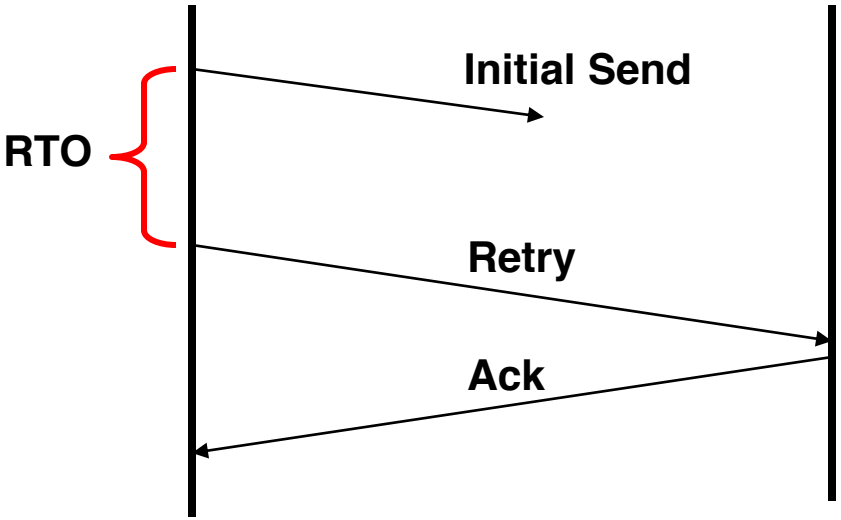
Flow Control Summary



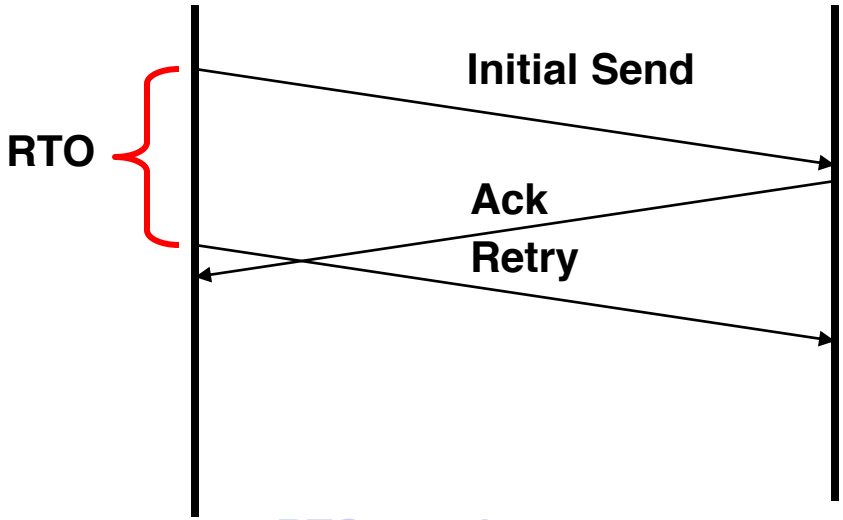
Transport layer: Setting Retransmission Timeouts

ECE 50863 – Computer Network Systems

Setting Retransmission Timeout (RTO)



Detect dropped packet



RTO too short

  - Time between sending & resending segment

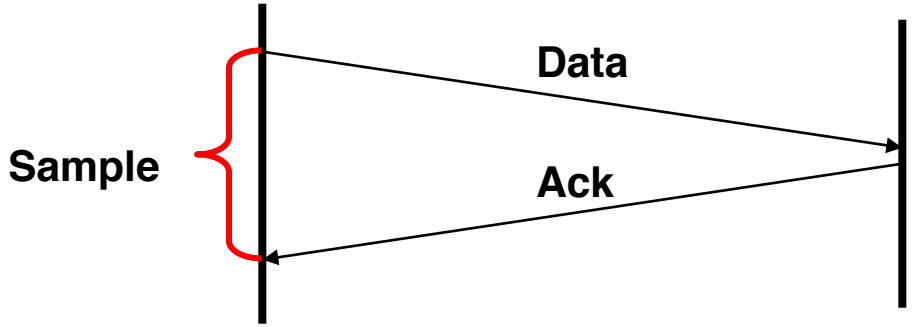
- Challenge
  - Too long: Add latency to communication when packets dropped
  - Too short: Send too many duplicate packets
  - General principle: Must be > 1 Round Trip Time (RTT)

Setting Timeouts

- Retransmission time-out should be set based on round-trip delay
- But round-trip delay different for each path!
- Must estimate RTT dynamically

Round-trip Time Estimation

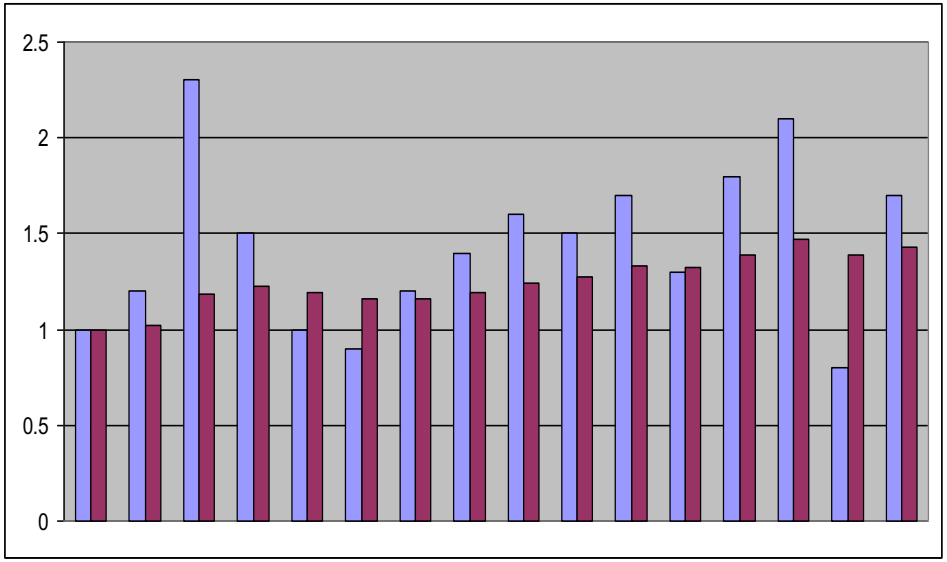
- Every Data/Ack pair gives new RTT estimate



- Can Get Lots of Short-Term Fluctuations

Original TCP Round-trip Estimator

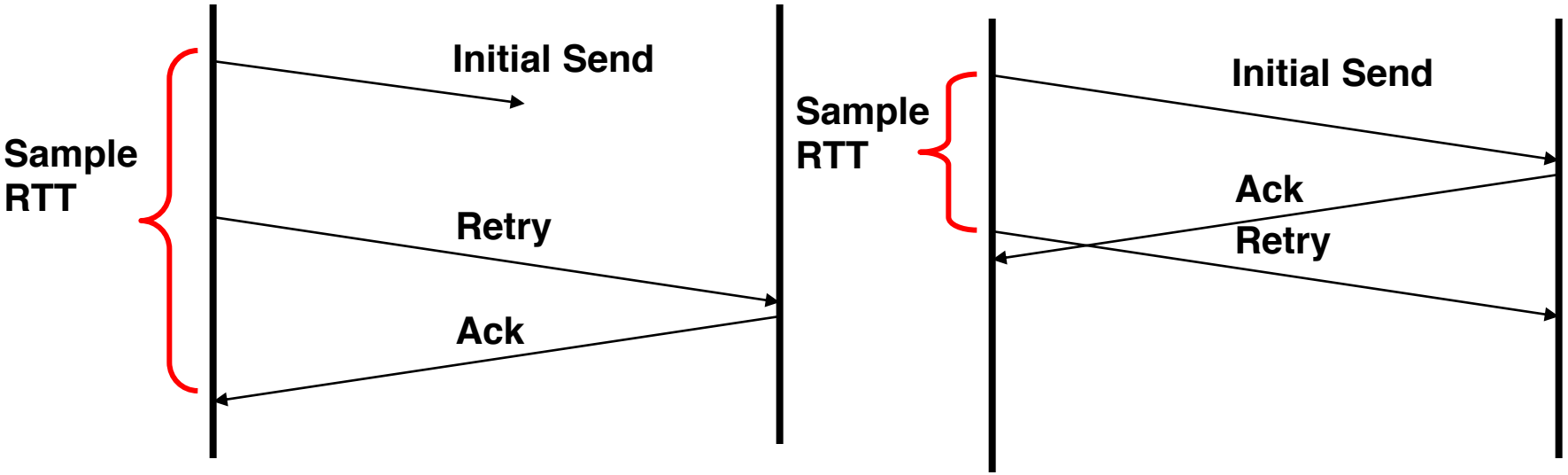
- Round trip times exponentially averaged:
  - New RTT = α (old RTT) + (1 - α) (new sample)
  - Recommended value for α: 0.8 - 0.9
    - 0.875 for most TCP's



- Retransmit timer set to (b * RTT), where b = 2
  - Every time timer expires, RTO exponentially backed-off

- Enhanced algorithm used in practice
  - [Jacobson/Karels: see textbook if interested]

Issue



What value of Sample RTT to use?

Solution: TCP simply neglects SampleRTT in such cases.
[Karn/Partridge algorithm]