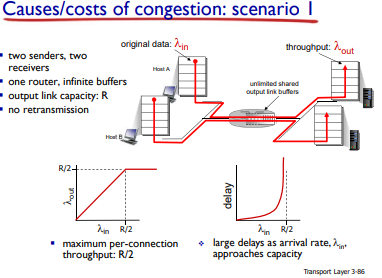
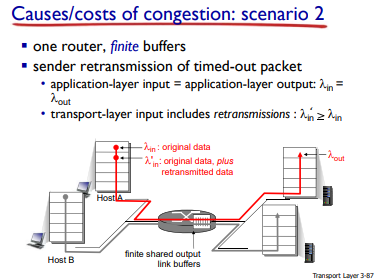
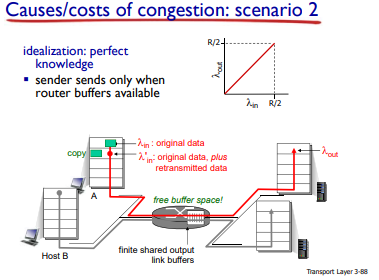
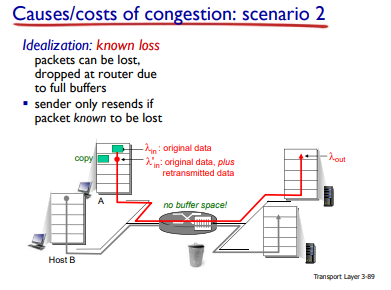
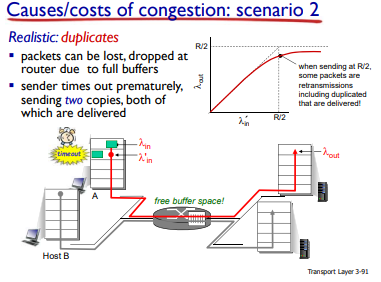
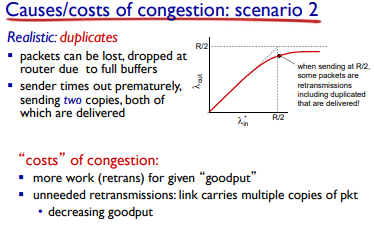
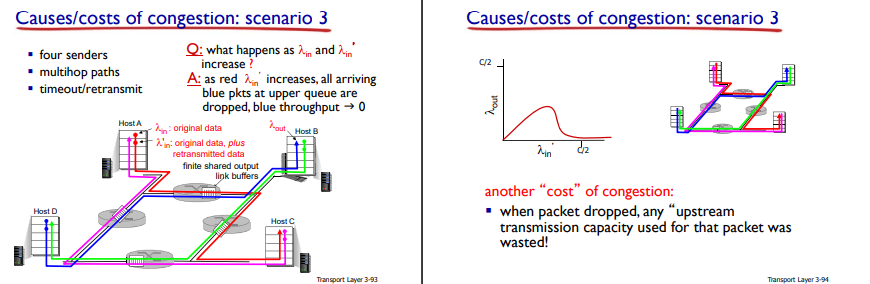
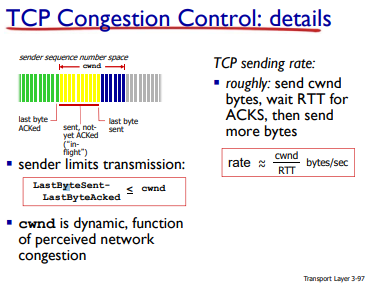
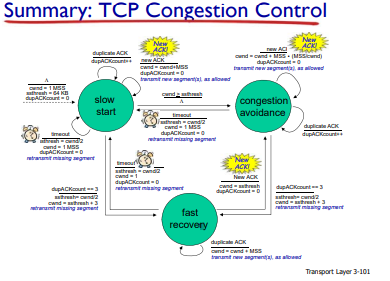
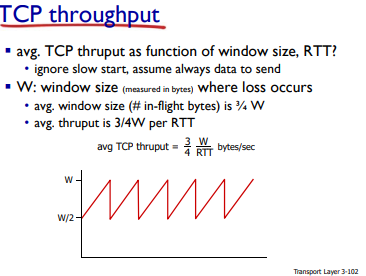
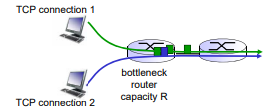
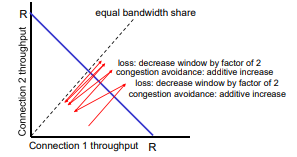
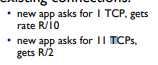
* Principles of Congestion Control
  + Congestion:
    - Informally: “too many source sending too much data too fast for a network to handle.”
    - Different from flow control
    - Manifestation:
      * Lost packets (buffer overflow at routers)
      * Long delays (queueing in router buffers)
  + Causes/Costs
    - 
  + 
  + 
  + 
  + 
  + 
  + Unneeded duplicates: packets not lost.
  + 
* TCP Congestion Control:
  + Additive increase multiplication decrease (AIMD)
  + Approach: sender increases transmission rate (window size) probing for usable bandwidth, until loss occurs
    - Additive increase: increase cwnd by 1 MSS every RTT until loss detected.
    - Multiplicative decrease: cut cwnd in half after loss.
* 
* TCP Slow Start
  + When connection begins, increase rate exponentially until first loss event:
    - Initially cwnd = 1 MSS  
      Double cwnd every RTT
    - Done by increment cwnd for every ACK received
  + SUMMARY: initial rate is slow but ramps up exponentially fast
* TCP: Detecting, reacting to loss
  + Loss indicated by timeout:
    - Cwnd set to 1 MSS;
    - Window then grows exponentially (as in slow start) to threshold, then grows linearly
  + Loss indicate by 3 duplicate ACKs; TCP RENO
    - Dup ACKs indidicate network capable of delivering some segments
    - Cwnd is cut in half window then grows linearly
  + TCP Tahoe always sets cwnd to 1 (timeout for 3 duplicate acks)
  + Implementation
    - Variable ssthresh
    - On loss event, ssthresh is set to ½ of cwnd just before loss event.
    - 
    - TCP Throughput
      * 
* TCP Fairness
  + Fairness goal: if K TCP sessions share same bottleneck link of bandwidth R, each should have average rate of R/K
  + 
  + Two Competing Sessions:
    - Additive increase gives slope of 1, as throughput increases
    - Multiplicative decrease decreases throughput proportionally
    - 
  + Fairness
    - Fairness and UDP
      * Multimedia apps often do not use TCP
        + Do not want rate throttled by congestion controls
      * Instead use UDP:
        + Send audio/video at constant rate, tolerate packet loss
    - Fairness, parallel TCP and Connections
      * Application can open multiple parallel connections between shots
      * Web browsers do this
      * E.g. link of rate R with 9 existing connections.
      * 
* Explicit Congestion Notification (ECN)
  + Network-assisted congestion control:
    - Two bits in IP header (ToS field) marke by network router to indicate congestion
    - Congestion indicated carried to receiving host
    - Receiver (seeing congestion indication in IP datagram)
      * Sets ECE bit to receiver-to-sender ACK segment to notify sender of congestion
      * 