CSc 450/550
Computer Communications & Networks
TCP over wireless links

Jianping Pan
(stand-in for Dr. Wu)

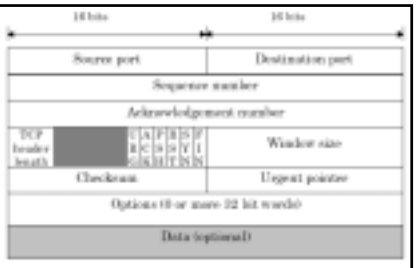
TCP over wireless links

- TCP
  - a quick review on how TCP works
- Wireless links
  - why wireless links are *so* different
- TCP over wireless links
  - performance issues
  - ways to improve TCP performance over wireless links

TCP/IP

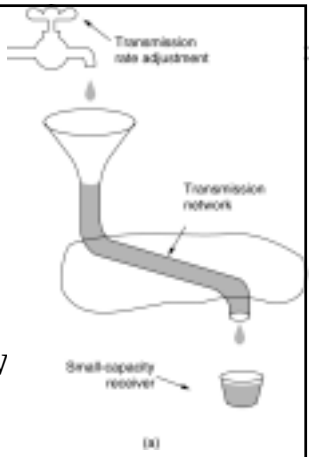
- TCP/IP
  - the Internet Protocol Suite
- TCP offers
  - reliable, in-sequence, stream-like data transfer
- IP offers
  - addressing hosts and routing packets
  - IP packets may be lost, corrupted, duplicated, reordered in the network

TCP



- Connection management
  - packet handshake (SYN, FIN, ACK)
- Flow, error, congestion control
  - sequence number
  - acknowledgment number
    - cumulative, delayed, duplicate acknowledgment
  - window size
  - checksum

Flow control

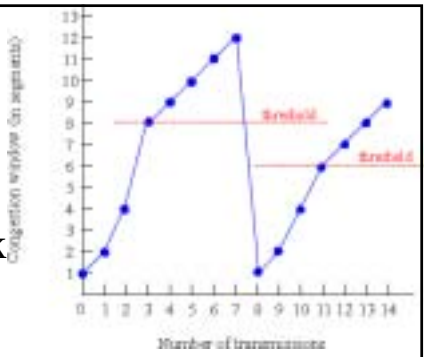


- Pace sender and receiver
  - according to their buffer size
- Receiver's (advertised) window
  - available buffer space
- Sender's window
  - sliding window
  - only send data within the window
  - sender's window < receiver's window

Error control

- Error detection
  - receiver: sequence number, TCP checksum
  - sender: timeout
- Error notification
  - receiver=>sender: duplicate acknowledgment
- Error recovery
  - sender: end-to-end retransmission

Congestion control



- Pace sender and network
- Slow start
- Congestion avoidance
  - ***assumption***: packet loss = network congestion
- Exponential back-off
- Fast retransmission
- Fast recovery

Wireless links

- Limited bandwidth
  - e.g., cellular systems
- Variable latency
  - e.g., WLAN
- Asymmetric link
  - e.g., Satellite
- Transmission error
  - due to noise, interference, mobility, etc

TCP over wireless

- Packet loss can be caused by
  - network congestion
  - transmission error (***more*** likely over wireless)
    - random error; burst error
- TCP is also affected by
  - limited bandwidth
  - variable latency
  - asymmetric link

Possible approaches

- Goal
  - react to different types of packet losses properly
- Hide link errors from TCP sender/receiver
  - at what cost?
- Let TCP sender/receiver figure out
  - how?
- Let someone tell TCP sender/receiver
  - who/when/how?

Link layer approach

- Forward error control (FEC) coding
- Link local acknowledgment and ARQ
- Protocol-aware link layer
  - e.g. access transport header to filter out dup packets
- Service-aware link layer
  - e.g. UCSC's TULIP - reliable service for TCP segment and ICMP message, unreliable service for UDP packets and TCP ack, piggyback short TCP packets in link-layer acknowledgment (half-duplex MAC)

End-to-end approach

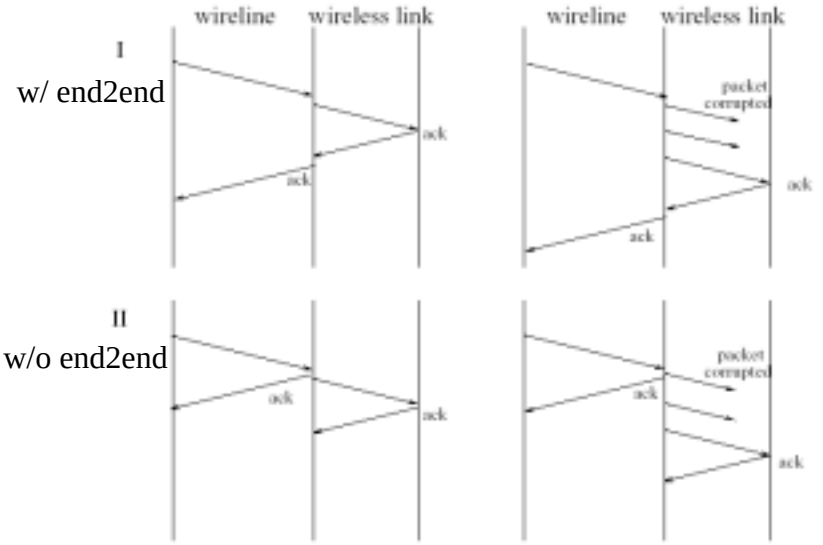
- End-to-end approach:
  - MITL's triggered fast retransmit (trade fast retransmit off timeout)
  - TAMU's delayed duplicated acknowledgment (give link layer more time to recover from error)
  - selective/negative acknowledgment
  - *sifest* sender (adjust *dupack* threshold according to amount of outstanding data)

End-to-end approach (2)

- End-to-end with implicit snooping:
  - UCB's Snoop (buffer data packet, local retransmission on local timeout or 1st dupack, filter out dupack if local retransmission is feasible)
  - m-TCP (keep last byte unacknowledged, acknowledge last byte if it's the last byte in a burst or when sever bad state is anticipated)
- End-to-end with explicit notification:
  - explicit bad state notification (ICMP-like message to reset sender timer), last hop acknowledgment (with lhack and receiver ack to discriminate congestion loss or corruption loss), freeze (advertise zero recv window)

Split connection

- Split connection approach
  - I-TCP (TCP over wired links, TCP or other protocol over wireless links, separated flow and error control), mobile-TCP (mobile gateway acts as transport agent for mobile), METP (also transport proxy)
- Split connection with end-to-end semantics
  - WTCP (retrain end-to-end acknowledgment)



This lecture

- TCP over wireless links
  - TCP overview: flow, error, congestion control
  - wireless links characteristics
  - TCP performance issues over wireless links
  - ways to improve TCP performance