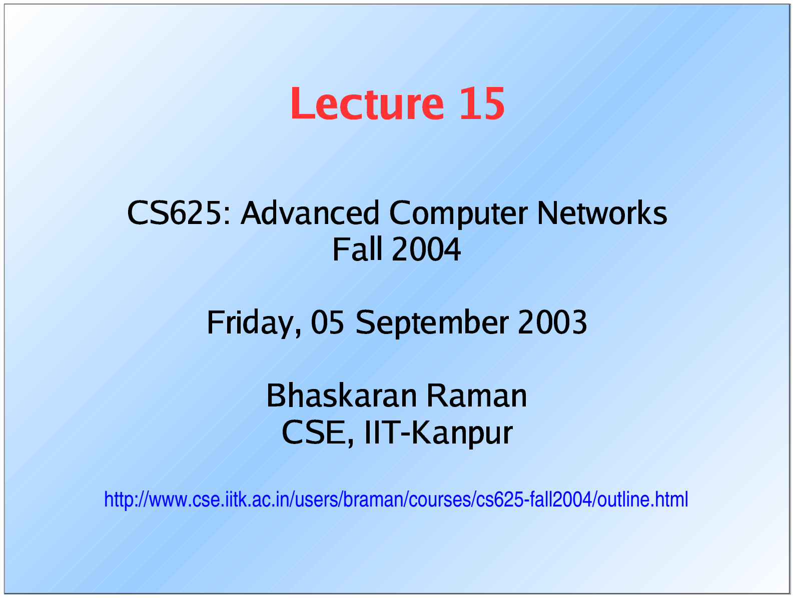


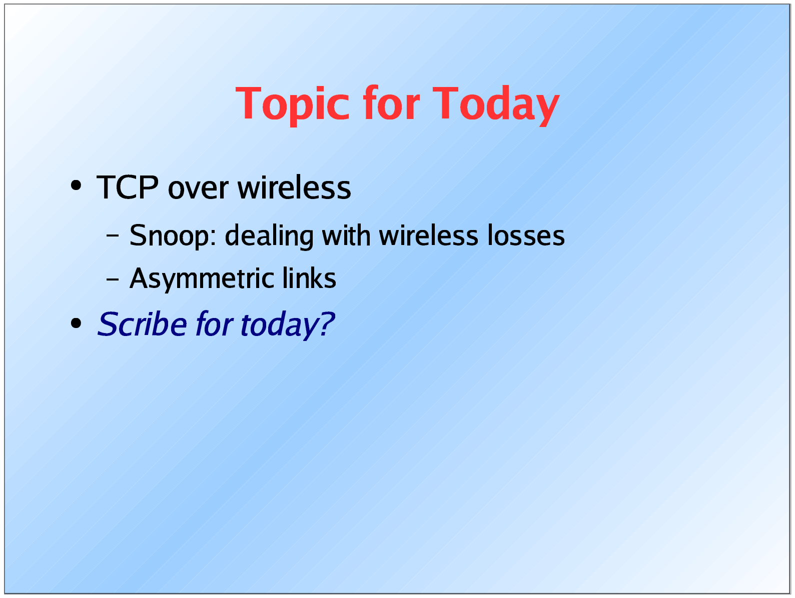
Lecture 15

CS625: Advanced Computer Networks
Fall 2004

Friday, 05 September 2003

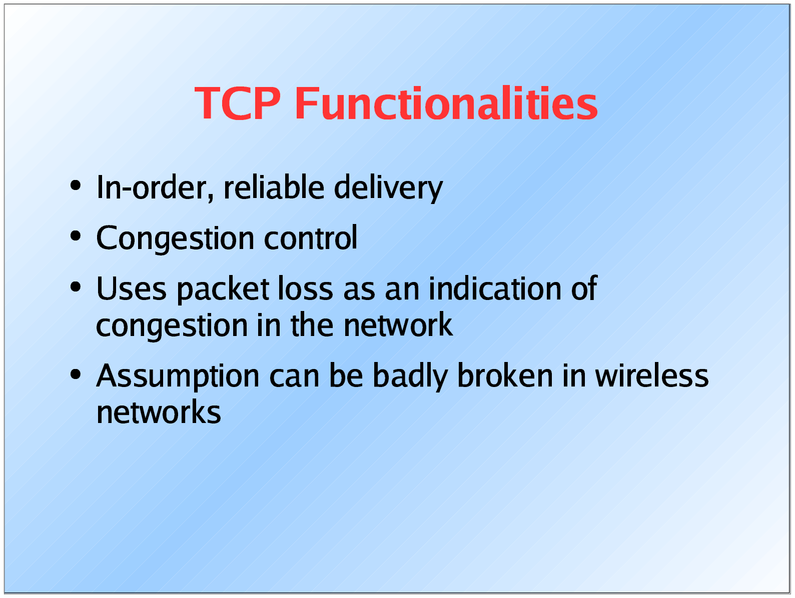
Bhaskaran Raman
CSE, IIT-Kanpur

http://www.cse.iitk.ac.in/users/braman/courses/cs625-fall2004/outline.html



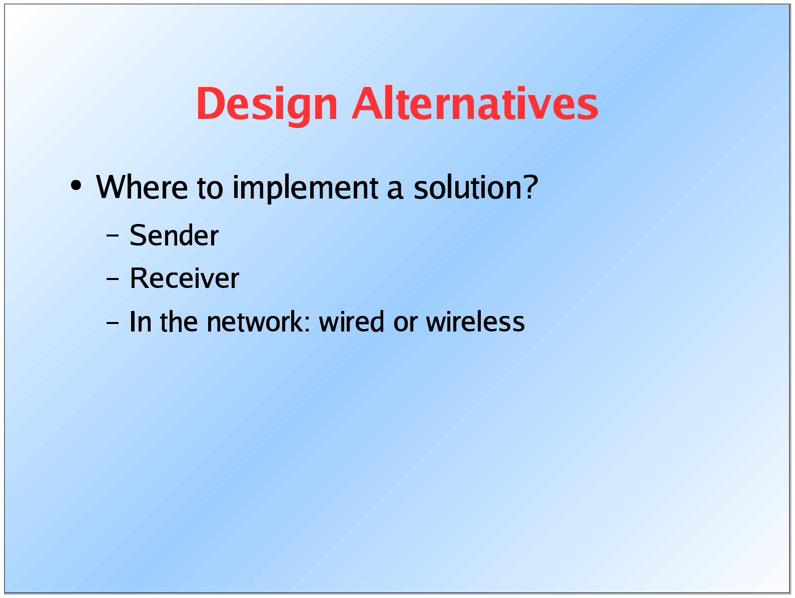
Topic for Today

- TCP over wireless
  - Snoop: dealing with wireless losses
  - Asymmetric links
- *Scribe for today?*



TCP Functionalities

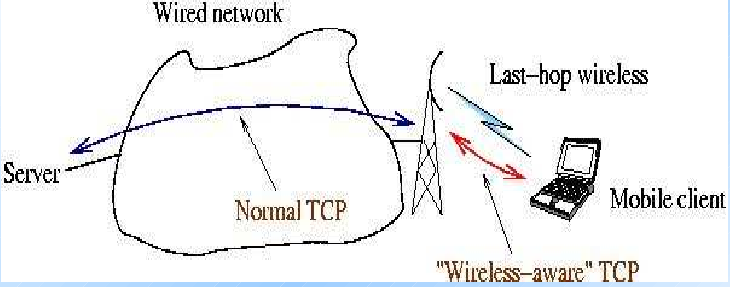
- In-order, reliable delivery
- Congestion control
- Uses packet loss as an indication of congestion in the network
- Assumption can be badly broken in wireless networks



Design Alternatives

- Where to implement a solution?
  - Sender
  - Receiver
  - In the network: wired or wireless

Design Alternatives (continued)



- Split-TCP
  - Base-station acts as a TCP proxy
  - Breaks end-to-end semantics
  - CPU/Memory overheads

Design Alternatives (continued)

- Reliable link layer
  - Have to be careful about link-layer timeout
- Snoop: network layer solution at base-station
  - Base-station caches packets as it forwards them
  - Detects packet loss, using DUP-ACKs
    - Does local retransmission
    - Suppresses DUP-ACKs from reaching server
    - snoop_data() and snoop_ack()

Details of snoop: snoop_data()

- Receiving packets (N-2), (N-1), N...
- In sequence (expected, or higher seq.no.)
  - Cache and send
- Out of sequence (seen already)
  - Ack not seen ==> forward
  - Ack seen ==> regenerate latest ACK
- Out of sequence (not seen earlier)
  - Record as a retransmit, cache, and forward

Details of snoop: snoop_ack()

- Receiving ACKs for (N-2), (N-1), N...
- New ACK ==> clear buffers, advance
- Spurious ACK ==> discard
- DUP-ACK
  - Packet not in cache ==>Forward on
  - Packet marked as retransmitted ==> Forward on
  - Packet in cache, first such DUP-ACK
    - Retransmit packet, suppress DUP-ACK, calculate expected number of further DUP-ACKs
  - Packet in cache, further DUP-ACKs
    - Suppress

Snoop: Some Remarks

- Will not work when IPSec is in place
- Link-layer solution achievable in many cases
- What method of evaluation?
  - Simulation
  - Implementation
  - Emulation

Asymmetric Network: Issues

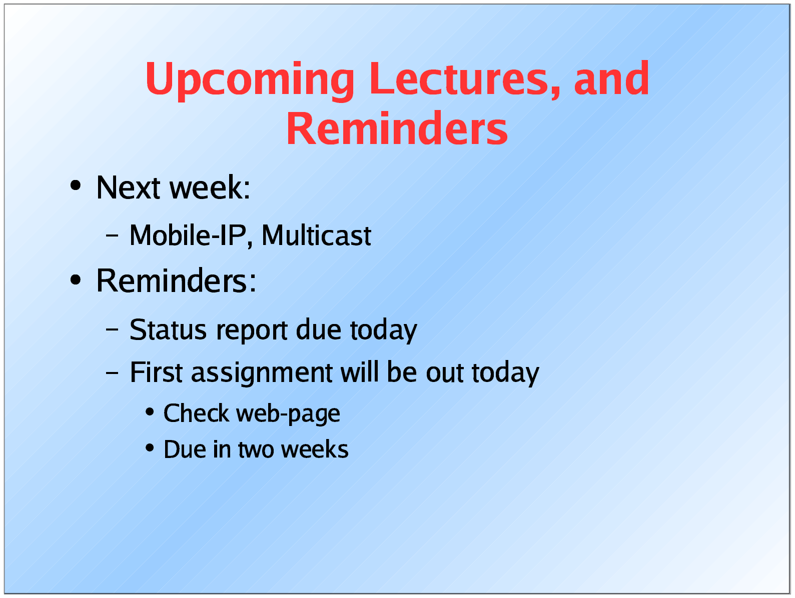
- Asymmetry in bandwidth
  - TCP ACK clocking mechanism breaks down
- ACK congestion
  - ACKs get spaced out
  - Normalized bandwidth ratio *k*
  - Only one ACK for *k* data packets can be sent
- ACK loss
  - Will slow down sender
  - Or cause sender burstiness
  - Will affect fast retransmit mechanism

Asymmetric Network: Solutions

- ACK Congestion Control:
  - If TCP receiver can learn of ACK loss, then it can send ACKs at a slower rate
  - Control delayed ACK parameter
- ACK filtering:
  - Congested router filters out “earlier” ACKs in its buffer
  - Does not require per-flow state in router

Asymmetric Network: Solutions (continued)

- Sender adaptation:
  - Send according to number of segments acknowledged, not based on number of ACKs
  - Calc. sending rate, and don't send too many back-to-back packets
  - Receiver marks DUP-ACKs with a bit, beyond threshold number of out-of-order packets
- Router can schedule ACKs with priority
  - Again, no per-flow state required



Upcoming Lectures, and Reminders

- Next week:
  - Mobile-IP, Multicast
- Reminders:
  - Status report due today
  - First assignment will be out today
    - Check web-page
    - Due in two weeks