Questions, so far?

- Android programming
- Team
- Project
From Principle to Practice:
Explore Wireless and Mobile Factors in Wireless Networks

Chunyi Peng
Fall 2015
Recap

Wireless networks

Access networks

Internet

Interconnection end-to-end layering: TCP/IP packet switched

... Edge last hop

Wireless broadcast Interference coverage ...

Mobility
Recap

• **Principles**
  – End-to-end argument
    • Vertical: Layering
    • Horizontal: end/edge, core

• **Two approaches**
  – Adaptation
    • Opportunistic
    • Model-based
  – Coordination
    • Cross-layer
    • Redirection (proxy)
**Agenda: From Principle to Practice**

- Using WiFi as an example
- Wireless TCP
  - How TCP varies in "wireless" networks
  - Source of loss varies -> TCP design varies
- EERA
  - How RA adapts to new requirements in "mobile" computing
  - Requirements on EE -> new RA design (not throughput driven)
Wireless Networks: WiFi

802.11 a/b/g/n/ac ...
Wireless networks, everywhere

- Campus
- Home
- Café
- Mall
- Bus stop
- Airport
- ...
- Hotspots, almost everywhere
IEEE 802.11 (WiFi) Standards

- In 1997, the Institute of Electrical and Electronic Engineers (IEEE) drafted the 802.11 standard for wireless local area networking (WLAN)
- 1999: 802.11b at 2.4GHz
- 1999: 802.11a at 5GHz
- 2003: 802.11g at 2.4GHz
- 2009: 802.11n at 2.4GHz, 5GHz, MIMO
- 2013: 802.11ac, wider channel
## 802.11 Standard: Toward Gigabit

<table>
<thead>
<tr>
<th>Network Standard</th>
<th>Maximum Speed (Mbps)</th>
<th>Range (meter)</th>
<th>Frequency (GHz)</th>
<th>Bandwidth (MHz)</th>
<th>Key Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.11b</td>
<td>11</td>
<td>30-50</td>
<td>2.4GHz</td>
<td>5</td>
<td>Modulation</td>
</tr>
<tr>
<td>802.11a</td>
<td>54</td>
<td>20-30</td>
<td>5 GHz</td>
<td>20</td>
<td>OFDM</td>
</tr>
<tr>
<td>802.11g</td>
<td>54</td>
<td>50-100</td>
<td>2.4GHz</td>
<td>20</td>
<td>OFDM</td>
</tr>
<tr>
<td>802.11n</td>
<td>Up to 600</td>
<td>Up to 100</td>
<td>2.4 GHz, 5 GHz</td>
<td>20, 40</td>
<td>MIMO</td>
</tr>
<tr>
<td>802.11ac</td>
<td>Up to 6.7Gbps</td>
<td>--</td>
<td>2.4, 5</td>
<td>20, 40, 80, 160</td>
<td>high-rate Modulation, MU-MIMO, Wider carrier</td>
</tr>
<tr>
<td>802.11ad</td>
<td>Up to 7Gps</td>
<td>1-10m</td>
<td>57-66 GHz (mmWave)</td>
<td>-</td>
<td>OFDM, Low power single carrier PHY,</td>
</tr>
</tbody>
</table>
Network architecture: Infrastructure Mode (AP)
Local Wireless Access

- Broadcast in nature
- #1: Collision
  - 802.11 MAC
  - Contention-based
  - Carrier sensing
  - Error detection and correction
Local Wireless Access

• Broadcast in nature
• #2: Interference
  – Resilient to varying channel/SNR
  – Rate adaptation
  – Interference cancellation
  – Beamforming (directional antenna)
  – MU-MIMO
In brief, major changes

• Wireless channel: more dynamic
• Bit error: more common
• Response time: longer (MAC overhead: sensing, backoff, retx)
• Cost for handling errors: higher

• Less secure (eavesdropping, jamming etc)
PROBLEM #1: WIRELESS TCP

Ref: most slides from Prof Lu’s CS211
Review: TCP Congestion Control

- Send as fast as possible, but not causing network congestion
  - Probe and adapt
  - AIMD: additive increase, multiplicative decrease
TCP congestion control over the Internet

• Premise
  – Packet loss is caused by congestion

• So, loss -> reducing sending rate
  – Cwnd (congestion window size) reduction
  – Timeout update
Issues for Wireless TCP

• Different packet loss behavior violates the assumption of TCP that all packet losses are due to congestion control:
  – congestion-induced loss: new flow joins, etc.
  – channel-error-induced loss: bursty or random channel error
  – handoff-induced packet loss: happens during handoff transition
  – routing-induced packet loss: stale routing tables (in a dynamic ad hoc network)

• “Uniform” reaction to different losses in TCP:
  – in TCP, reduce congestion window by half upon packet loss
  – Does “one-fit-all” work in the wireless scenario?
So,
TCP does not work well with wireless!

We need wireless TCP, but HOW?
Wireless TCP

• Design space and constraints
  – when to use?
  – which choice?

• Understanding of
  – TCP snoop
  – Other design options: ECN, WTCP

• State-of-the-art
  – Data-driven approach (Learnability), SIGCOMM’13, SIGCOMM’15
Solution Constraints

• Where you are allowed to modify/add the design, what is the information you can get
  
  – Scenario 1: **no change at any intermediate node** (e.g. base station), change is only allowed at both the sender & receiver sides; however, wireless link may provide information regarding whether it is wireless-related loss or not

  – Scenario 2: **no change at any intermediate node, no change at the receiver side**, only sender side is allowed to be modified; no addition information feedback other than loss
Solution Constraints

– Scenario 3: Intermediate node is allowed to change, **TCP sender & receiver sides should be kept intact** as much as possible

– Scenario 4: you can **change anywhere**, TCP senders & receivers, as well as intermediate nodes
Solution

• **Split-connection approach**
  – split into two connections: one for wireline part, the other for wireless part
  – since the wireless connection is only one hop, much easier to handle (link layer solution is possible)

• **Link-layer solution**
  – provide higher layer a logical “lossless” link, by using link layer techniques for loss recovery: retransmissions, FEC
  – shield TCP from wireless loss as much as possible
Split-Connection Schemes

• Split a single end-to-end TCP connection at an intermediate node (typically a base station), maintain two separate TCP connections for wireline links and for wireless links

• Comments:
  – duplicate data copying across layers (there are optimization techniques for sure)
  – even if wireless TCP is a single hop, it is still TCP, you still need to modify it to perform properly over the wireless link
  – what do you gain anyway?
Link-Layer Solutions

• Play with different link-layer recovery schemes by recovering loss locally:
  – ARQ, FEC, channel-swapping, hybrid

• suppress DUP_ACKs from TCP to some degree

• how much you can shield from the standard TCP?
TCP Snoop for cellular networks

• Cache packets at the base station and perform local retransmissions across the wireless link

• Snoop data packets:
  – A new packet in the normal TCP sequence: add to the snoop cache, forward to MH
  – Out-of-sequence packet that cached earlier (compare its seq # with latest ACK#): sender timeouts or fast retransmit
  – Out-of-sequence packet that NOT cached: congestion or out-of-order delivery, forwarded to MH and marked as rexmitted by sender

• Snoop ACK:
  – A new ACK: flush the cached acked data packets, update RTT estimate, propagate ACK to sender
  – A spurious ACK (seq #< last ACK): discarded
  – DUP ack: (a) if not 1st DUP ack (later dup acks for lost packets), discard; (b) 1st DUP ack, retransmit lost packet with priority.
Solution: End-to-End Protocol (contd’)

• Category 1:
  – the network or receiver provides additional feedback information on the loss behavior
  – enhance TCP to handle bursty loss using SACK

• Category 2:
  – at the sender side, develop mechanisms differentiating packet loss behaviors, modify TCP congestion control to react accordingly
End to end schemes: Category 1

- Use explicit loss notification or ECN to differentiate loss behaviors
  - upon ELN, retransmit lost packet, do not invoke congestion control algorithm
  - an alternative scheme: invoke congestion control only upon ECN
  - comments: what about severe wireless loss (indeed the effective channel capacity decreases at the moment), treat it as congestion or non-congestion?

- TCP only reacts to ECN or ELN signals
- Use TCP SACK to handle multiple back-to-back loss
End-to-end Schemes: Category 2

• Goals:
  – maintain TCP like congestion control mechanism, in particular AIMD principle
  – basic design relies on packet loss indication only
  – able to better adapt if network can provide additional information feedback
  – differentiating different packet loss behaviors: congestion induced loss, channel error induced loss, (capacity) probe loss, etc.
  – tailor rate adaptation to different loss behaviors
Illustration of expected behaviors
End-to-end Schemes: Category 2

- use history of packet loss and congestion profile to distinguish congestion-induced loss from non-congestion-induced loss
  - maintain moving average & deviation for effective transmission rate
  - check whether the current is within the deviation

- reacts gently to non-congestion-induced loss and reacts aggressively to congestion-induced loss, thereby achieving high efficiency, as well as quick response to the onset of congestion
  - gentle loss: small decrease factor
  - heavy loss: exponential growth for decrease factor
WTCP (FYI)

• End-to-end design for very-low speed wide-area wireless network
  – CDPD

• Congestion control
  – Rate-based design
  – Inter-packet delay as the main mechanism for transmission control
  – Distinguish the cause of packet loss and adjust the transmission rate accordingly
  – transmission rate computation at the receiver
Packet Loss Detection Approaches

• Network Oriented Approaches
  – ELFN, G.Holland and N.Vaidya, Mobicom’ 99

• End-to-end Approaches
  – Use packet inter-arrival time to detect congestion
    S.Biaz and N.Vaidya IEEE ASSET’ 99
  – Use out-of-order delivery to detect route change
    F.Wang and Y.Zhang, Mobihoc’ 02
Measurement Noise In End-to-End Approach

- Detections based on End host measurement can be very inaccurate
  - By noise, we mean measurements leading to false detections
  - For example, use packet delay to detect congestion, we expect:
    - High delay measurement \(\rightarrow\) congestion
    - Low delay measurement \(\rightarrow\) non-congestion
  - Several noise factors due to dynamic network conditions
    - Large variance in packet delay or out-of-order delivery
    - Random losses
    - Bursty traffic source
Improving End-to-End Detection – Problem Statement

• Low delay during congestion is rare
  • Delay dominated by prolonged queuing
• High delay during non-congestion is frequent
  – In non-congestion state, noise factors contribute to high delay
• False congestion alarms during non-congestion state dominate detection inaccuracy
• The problem statement
  – Reduce the false congestion alarm in non-congestion states
  – Maintaining responsiveness to network congestion
Summary

• **Change factors**: wireless loss + delay

• **Problem**
  – TCP for Internet: *success*
  – TCP for Internet over wireless: *perform badly*

• **Challenges and opportunities**
  – Distinguish congestion from wireless loss
  – Information from lower-layers (detection possible)

• **Solution (ideas)**
  – Split TCP over wireline and wireless (wireless: edge)
  – Detect wireless behaviors (loss or congestion?)
IN-CLASS PRESENTATION

An Experimental Study of the Learnability of Congestion Control, SIGCOMM'13
Adaptive Congestion Control for Unpredictable Cellular Networks, SIGCOMM’15