Acknowledgments: Lecture slides are from the graduate level Computer Networks course thought by Srinivasan Seshan at CMU. When slides are obtained from other sources, a reference will be noted on the bottom of that slide. A full list of references is provided on the last slide.
Wireless Intro

• TCP on wireless links
• Wireless MAC
• Assigned reading
  • [BPSK97] A Comparison of Mechanism for Improving TCP Performance over Wireless Links
  • [BM09] In Defense of Wireless Carrier Sense
Wireless Challenges

• Force us to rethink many assumptions
• Need to share airwaves rather than wire
  • Don’t know what hosts are involved
  • Host may not be using same link technology
• Mobility
• Other characteristics of wireless
  • Noisy $\rightarrow$ lots of losses
  • Slow
  • Interaction of multiple transmitters at receiver
    • Collisions, capture, interference
  • Multipath interference
Overview

• Wireless Background

• Wireless MAC
  • MACAW
  • 802.11

• Wireless TCP
Transmission Channel Considerations

- Every medium supports transmission in a certain frequency range.
  - Outside this range, effects such as attenuation, .. degrade the signal too much
- Transmission and receive hardware will try to maximize the useful bandwidth in this frequency band.
  - Tradeoffs between cost, distance, bit rate
- As technology improves, these parameters change, even for the same wire.
  - Thanks to our EE friends
The Nyquist Limit

- A noiseless channel of width H can at most transmit a binary signal at a rate $2 \times H$.
  - E.g. a 3000 Hz channel can transmit data at a rate of at most 6000 bits/second
  - Assumes binary amplitude encoding
Past the Nyquist Limit

• More aggressive encoding can increase the channel bandwidth.
  • Example: modems
    • Same frequency - number of symbols per second
    • Symbols have more possible values

\[ \text{psk} \]

\[ \text{Psk} + \text{AM} \]
• Can’t add infinite symbols - you have to be able to tell them apart. This is where noise comes in.

• Shannon’s theorem:
  • $C = B \times \log(1 + S/N)$
  • $C$: maximum capacity (bps)
  • $B$: channel bandwidth (Hz)
  • $S/N$: signal to noise ratio of the channel
    • Often expressed in decibels (db). $10 \log(S/N)$

• Example:
  • Local loop bandwidth: 3200 Hz
  • Typical $S/N$: 1000 (30db)
  • What is the upper limit on capacity?
    • Modems: Teleco internally converts to 56kbit/s digital signal, which sets a limit on $B$ and the $S/N$. 
Cellular Reuse

- Transmissions decay over distance
  - Spectrum can be reused in different areas
  - Different "LANs"
  - Decay is $1/R^2$ in free space, $1/R^4$ in some situations
Multipath Effects

• Receiver receives multiple copies of the signal, each following a different path.

• Copies can either strengthen or weaken each other.
  • Depends on whether they are in or out of phase.

• Small changes in location can result in big changes in signal strength.
  • Short wavelengths, e.g. 2.4 GHz → 12 cm.

• Difference in path length can cause inter-symbol interference (ISI).
Fading - Example

- Frequency of 910 MHz or wavelength of about 33 cm
Overview

- Wireless Background

- Wireless MAC
  - MACAW
  - 802.11

- Wireless TCP
Medium Access Control

• Think back to Ethernet MAC:
  • Wireless is a shared medium
  • Transmitters interfere
  • Need a way to ensure that (usually) only one person talks at a time.
    • Goals: Efficiency, possibly fairness
Example MAC Protocols

• Pure ALOHA
  • Transmit whenever a message is ready
  • Retransmit when ACK is not received

• Slotted ALOHA
  • Time is divided into equal time slots
  • Transmit only at the beginning of a time slot
  • Avoid partial collisions
  • Increase delay, and require synchronization

• Carrier Sense Multiple Access (CSMA)
  • Listen before transmit
  • Transmit only when no carrier is detected
CSMA/CD Does Not Work

- Carrier sense problems
  - Relevant contention at the receiver, not sender
  - Hidden terminal
  - Exposed terminal
- Collision detection problems
  - Hard to build a radio that can transmit and receive at same time
MACA (RTS/CTS)

RTS = Request-to-Send

assuming a circular range
MACA (RTS/CTS)

RTS = Request-to-Send

NAV = remaining duration to keep quiet

NAV = 10
MACA (RTS/CTS)

CTS = Clear-to-Send
MACA (RTS/CTS)

CTS = Clear-to-Send

NAV = 8
MACA (RTS/CTS)

• **DATA** packet follows CTS. Successful data reception acknowledged using **ACK**.
MACA (RTS/CTS)

Reserved area
MACAW: Additional Design

- ACK (needed for faster TCP transfers)

<table>
<thead>
<tr>
<th>Error Rate</th>
<th>RTS-CTS-DATA</th>
<th>RTS-CTS-DATA-ACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>40.41</td>
<td>36.76</td>
</tr>
<tr>
<td>0.001</td>
<td>36.58</td>
<td>36.67</td>
</tr>
<tr>
<td>0.01</td>
<td>16.65</td>
<td>35.52</td>
</tr>
<tr>
<td>0.1</td>
<td>2.48</td>
<td>9.93</td>
</tr>
</tbody>
</table>

- DS (needed since carrier sense disabled)

A: CTS
B: RTS
C: Hears RTS

Doesn’t hear CTS
Hears DS
Overview

• Wireless Background

• Wireless MAC
  • MACAW
  • 802.11

• Wireless TCP
802.11 particulars

- **802.11b (WiFi)**
  - Frequency: 2.4 - 2.4835 Ghz DSSS
  - Rates: 1, 2, 5.5, 11 Mbps
- **802.11a**: Faster, 5Ghz OFDM. Up to 54Mbps, 19+ channels
- **802.11g**: Faster, 2.4Ghz, up to 54Mbps
- **802.11n**: 2.4 or 5Ghz, multiple antennas (MIMO), up to 450Mbps (for 3x3 antenna configuration)
Overview, 802.11 Architecture

BSS: Basic Service Set
ESS: Extended Service Set
802.11 modes

- **Infrastructure mode**
  - All packets go through a base station
  - Cards associate with a BSS (basic service set)
  - Multiple BSSs can be linked into an Extended Service Set (ESS)
    - Handoff to new BSS in ESS is pretty quick
      - Wandering around CE building
    - Moving to new ESS is slower, may require re-addressing
      - Wandering from Sharif to Tehran U.

- **Ad Hoc mode**
  - Cards communicate directly.
802.11 Management Operations

- Scanning
- Association/Reassociation
- Time synchronization
- Power management
Scanning & Joining

• Goal: find networks in the area

• Passive scanning
  • No require transmission → saves power
  • Move to each channel, and listen for Beacon frames

• Active scanning
  • Requires transmission → saves time
  • Move to each channel, and send Probe Request frames to solicit Probe Responses from a network
Association in 802.11

1: Association request
2: Association response
3: Data traffic
Time Synchronization in 802.11

- Timing synchronization function (TSF)
  - AP controls timing in infrastructure networks
  - All stations maintain a local timer
  - TSF keeps timer from all stations in sync

- Periodic Beacons convey timing
  - Beacons are sent at well known intervals
  - Timestamp from Beacons used to calibrate local clocks
  - Local TSF timer mitigates loss of Beacons
Power Management in 802.11

- A station is in one of the three states
  - Transmitter on
  - Receiver on
  - Both transmitter and receiver off (dozing)
- AP buffers packets for dozing stations
- AP announces which stations have frames buffered in its Beacon frames
- Dozing stations wake up to listen to the beacons
- If there is data buffered for it, it sends a poll frame to get the buffered data
802.11 DCF ([RTS/CTS/]Data/ACK)

Station 1
- NAV
- random backoff (7 slots)
- SIFS
- CTS
- SIFS
- ACK
- new random backoff (10 slots)
- Station defers
- NAV

Station 2
- NAV
- DIFS
- RTS
- SIFS
- DATA
- DIFS
- Station defers

Station 3
- NAV
- DIFS
- random backoff (9 slots)

Station 4
- NAV
- DIFS
- Station defers, but keeps backoff counter (=2)
- DIFS
- Station sets NAV upon receiving RTS
- DATA
- SIFS

Station 5
- NAV
- SIFS
- ACK

Station 6
- NAV
- DATA
- SIFS
- Station sets NAV upon receiving RTS
- Station sets NAV upon receiving CTS, this station is hidden to station 1
Discussion

• RTS/CTS/Data/ACK vs. Data/ACK
  • Why/when is it useful?
  • What is the right choice
  • Why is RTS/CTS not used?
802.11 Rate Adaptation

• 802.11 spec specifies rates not algorithm for choices
  • 802.11b 4 rates, 802.11a 8 rates, 802.11g 12 rates
  • Each rate has different modulation and coding

Transmission Rate ↑ then Loss Ratio ↑
Transmission Rate ↓ then Capacity Utilization ↓

throughput decreases either way – need to get it just right
Carrier Sense

Desired result: concurrency

Desired result: time-multiplexing

Desired result: ???
Single Receiver, Sender and Interferer

\[ D = 55 \]

- [ ] Prefers concurrency
- [ ] Prefers multiplexing
- [ ] Starved w/o multiplexing
Interferer Position

Receiver preference vs. position:

**Excellent agreement on multiplexing**

**Disagreement??**

**Excellent agreement on concurrency**

- **S** prefers concurrency
- **I** prefers multiplexing
- **Starved w/o multiplexing**

D = 20

D = 55

D = 120
ABR Helps in Disagreements

• Intermediate distance can mean poor agreement! But...
• Does “mistaken” multiplexing mean 50%-reduced throughput? No. Adapts with higher bitrate.
Carrier Sense + ABR Works Well

![Graph showing the relationship between Fraction of throughput and S-I distance (D). The graph includes lines for Optimal, Multiplexing, Concurrency, and Carrier Sense (D_{thresh} = 55) with an indication that Inefficiency is small. The maximum value of R is 55.](image)
Key Assumptions

• ABR == Shannon
  • ABR is rarely this good

• Interference and ABR are both stable
  • Interference may be bursty/intermittent
Overview

• Wireless Background

• Wireless MAC
  • MACAW
  • 802.11

• Wireless TCP
Wireless Challenges

- Force us to rethink many assumptions
- Need to share airwaves rather than wire
  - Don’t know what hosts are involved
  - Host may not be using same link technology
- Mobility
- Other characteristics of wireless
  - Noisy → lots of losses
  - Slow
  - Interaction of multiple transmitters at receiver
    - Collisions, capture, interference
  - Multipath interference
TCP Problems Over Noisy Links

• Wireless links are inherently error-prone
  • Fades, interference, attenuation
  • Errors often happen in bursts

• TCP cannot distinguish between corruption and congestion
  • TCP unnecessarily reduces window, resulting in low throughput and high latency

• Burst losses often result in timeouts
• Sender retransmission is the only option
  • Inefficient use of bandwidth
Constraints & Requirements

- Incremental deployment
  - Solution should not require modifications to fixed hosts
  - If possible, avoid modifying mobile hosts
- Probably more data to mobile than from mobile
  - Attempt to solve this first
Challenge #1: Wireless Bit-Errors

Burst losses lead to coarse-grained timeouts
Result: Low throughput
Performance Degradation

Best possible TCP with no errors (1.30 Mbps)

TCP Reno (280 Kbps)

2 MB wide-area TCP transfer over 2 Mbps Lucent WaveLAN
Proposed Solutions

• End-to-end protocols
  • Selective ACKs, Explicit loss notification

• Split-connection protocols
  • Separate connections for wired path and wireless hop

• Reliable link-layer protocols
  • Error-correcting codes
  • Local retransmission
Approach Styles (End-to-End)

- Improve TCP implementations
  - Not incrementally deployable
  - Improve loss recovery (SACK, NewReno)
  - Help it identify congestion (ELN, ECN)
    - ACKs include flag indicating wireless loss
  - Trick TCP into doing right thing → E.g. send extra dupacks
- What is SMART?
  - DUPACK includes sequence of data packet that triggered it
Approach Styles (Split Connection)

• Split connections
  • Wireless connection need not be TCP
  • Hard state at base station
    • Complicates mobility
    • Vulnerable to failures
    • Violates end-to-end semantics
Split-Connection Congestion Window

- Wired connection does not shrink congestion window
- But wireless connection times out often, causing sender to stall
Approach Styles (Link Layer)

- More aggressive local retransmit than TCP
  - Bandwidth not wasted on wired links
- Adverse interactions with transport layer
  - Timer interactions
  - Interactions with fast retransmissions
  - Large end-to-end round-trip time variation
- FEC does not work well with burst losses
Hybrid Approach: Snoop Protocol

- Shield TCP sender from wireless vagaries
  - Eliminate adverse interactions between protocol layers
  - Congestion control only when congestion occurs
- The End-to-End Argument [SRC84]
  - Preserve TCP/IP service model: end-to-end semantics
- Eliminate non-TCP protocol messages

Fixed to mobile: transport-aware link protocol
Mobile to fixed: link-aware transport protocol
Snoop Overview

- Modify base station
  - to cache un-acked TCP packets
  - ... and perform local retransmissions

- Key ideas
  - No transport level code in base station
  - When node moves to different base station, state eventually recreated there
Snoop Protocol: CH to MH

- **Snoop agent**: *active interposition agent*
  - Snoops on TCP segments and ACKs
  - Detects losses by duplicate ACKs and timers
  - Suppresses duplicate ACKs from MH
Snoop Protocol: CH to MH

- Transfer of file from CH to MH
- Current window = 6 packets
Snoop Protocol: CH to MH

- Transfer begins
• Snoop agent caches segments that pass by
Snoop Protocol: CH to MH

- Packet 1 is Lost
Snoop Protocol: CH to MH

- Packet 1 is Lost
  - Duplicate ACKs generated
Packet 1 is Lost
- Duplicate ACKs generated
- Packet 1 retransmitted from cache at higher priority
Snoop Protocol: CH to MH

- Duplicate ACKs suppressed
Snoop Protocol: CH to MH

- Clean cache on new ACK
Snoop Protocol: CH to MH

- Clean cache on new ACK
Snoop Protocol: CH to MH

- Active soft state agent at base station
- Transport-aware reliable link protocol
- Preserves end-to-end semantics
Performance: FH to MH

Throughput (Mbps)

1/Bit-error Rate (1 error every x Kbits)

- Snoop+SACK and Snoop perform best
- Connection splitting not essential
- TCP SACK performance disappointing

2 MB local-area TCP transfer over 2 Mbps Lucent WaveLAN
Discussion

• Real link-layers aren’t windowed
  • Out of order delivery not that significant a concern

• TCP timers are very conservative