Wireless and Mobile Networking

CSU CS557, Spring 2018
Instructor: Lorenzo De Carli
(Slides by Christos Papadopoulos, remixed by Lorenzo De Carli)
Overview

• Wireless access and mobility
  – force us to rethink many of our assumptions

• Class focus:
  – Link layer issues (MACA/MACAW)
  – Ad-hoc mobile wireless networks (DSR)
Dimensions of Wireless/Mobile

- **topology**
  - one-hop to base-station vs. ad hoc/multi-hop

- **mobility**
  - fixed vs. mobile

- **protocols**
  - IP vs. cell phone (3G, 4G, 5G, etc.)

- **constraints:**
  - energy
  - radio range
  - antenna directionality

- **trust**
  - do you trust others to forward your data and overhear your packets?

- **app-level issues**
  - even if you have connectivity, what can you do?
    - ex. it may be easier to share files with USB drives!
    - what more to do besides connect to wired Internet?
Radio Propagation Reality

- Reality is *much* worse
- Slow fading
  - Propagation path loss can be different in different directions
  - Caused by environmental artifacts
- Fast fading
  - Interfering multi-path transmissions

(data from Jerry Zhao, ISI, 2002)
MACAW: A Media Access Protocol for Wireless LANs

Bharghavan94a
The Physical Layer

• Based on MACA
  – Multiple Access Collision Avoidance protocol

• First study a simple model
  – radio transmission range defined by cell
  – a receiver within range can hear transmission
Radio Propagation

• The physics:
  – Radios transmit at certain power
  – Received reception ability falls off as $r^{-4}$
  – If signal to noise ratio is high-enough, then receiver can detect transmission

• Other, more complex environmental interactions:
  – multi-path: reflected signals interfere with original
  – Also, transmitters can interact negatively with each other (multiple overlapping signals may degrade each other to the point where no signal can be reconstructed!)
Node capabilities

- Node can either transmit or receive
- Node cannot perform carrier sensing (i.e. check whether channel is busy)
- Very limited assumptions about hardware capabilities (keeps the hardware cheap!)
Interaction between transmitters

- Interactions of multiple transmitters at receiver
  - **Collision**: if receiver is within range of two transmitters, but can’t extract either
  - **Capture**: one signal stronger than other
  - **Interference**: in-range of one transmitter, out of range of another, but can’t extract signal

- Other, more complex environmental interactions
  - multi-path: reflected signals interfere with original
Interaction between transmitters - II

- Various effects
  - Collisions
  - Capture
  - Interference

A and B transmit simultaneously
Carrier Sense Does Not Work

- **Carrier Sense**: before transmitting, check if carrier present
  - Used in the wired Ethernet of yore, when multiple hosts could share the same medium (e.g. using hubs)
  - Makes sense in Ethernet: all hosts share the same medium, if one is transmitting eventually all others will detect it

- **Wireless is different!**
  - Relevant contention at the receiver, not sender
  - Contention happening at the receiver may be such that the sender will never know about it!
• **Hidden terminals**
  • A and C transmit to B
  • Collision occurs at B, so B cannot extract any signal
  • From the point of view of A, C is an hidden terminal
  • Carrier sense at A is pointless since A cannot sense C’s transmission anyway
Carrier Sense Does Not Work - II

- **Exposed terminals**
  - B’s transmission inhibits C from transmitting to D
  - C is exposed to transmission/interference from B
  - However, C could transmit to D since A is out of range of C
  - Carrier sense at C leads to unnecessarily conservative decision
Better Approach

• Before sending data, send Ready-to-Send (RTS) with length of data
  – Any station that hears RTS defers transmission

• Target responds with Clear-to-Send (CTS) echoing length of data
  – CTS contains length of data
  – Any station that hears CTS defers long enough for data transmission to complete

• RTS/CTS implemented in MACA (Karn 1990)
How does RTS/CTS solve hidden and exposed terminals?

Hidden terminal (A and C both transmitting to B): C cannot hear A’s RTS, but can hear B’s CTS and defer transmission.

Exposed terminal (C being prevented from transmitting by B transmitting to A): C hears B’s RTS but not A’s CTS, so it can infer that its transmission would not affect A.
  - But still not really solved! C can send RTS but not hear CTS.
  - Will talk about this later.
Better Approach - III

- Deferrals solve the hidden and exposed terminal scenario in the right way

- If CTS is not heard, or RTS collides
  - retransmit RTS after binary exponential back-off

- We’re not done yet
Back-off Woes

• Backoff algorithm
  – Keep a backoff counter
  – Choose backoff uniformly between 0 and backoff counter
  – Decrease counter when CTS is heard
  – Increase when not
  – Most common: binary exponential backoff

• Makes sense intuitively - but creates unfairness!
Back-off Woes - II

- Backoff algorithm can lead to unfairness
  - If one node starts off with a high value of counter, it can be starved
- Example: both B and C contend to transmit to A
  - B wins, C loses and increases its backoff counter
  - Next round, C is more likely to lose again due to its higher backoff counter
  - C loses again, increasing its backoff counter
  - ...
Back-off Woes - III

- What is the fundamental issue?
  - Backoff counter should reflect congestion level in the network
  - Congestion level is the same for all nodes!
- Fix: Need to share congestion information
  - backoff copy: nodes copy backoff counters from other nodes’ transmissions
  - Everyone has the same backoff counter - everyone has the same fair chance
- Other optimization: Avoid oscillations in backoff counter
  - increase multiplicatively
  - decrease additively
  - improves throughput
Adding Reliability

- Wireless losses possible due to noise or collisions
- Add an ACK after DATA transmission
  - if ACK not received, sender restarts RTS/CTS again
  - if ACK was lost, receiver sends ACK instead of CTS
  - Back-off counter increased if ACK not received, decreased otherwise
- This is a textbook example of why violations to end-to-end principle may be acceptable to increase performance
Fairness I

• RTS/CTS does not really solve exposed terminal: C can hear B’s RTS but:
  – doesn’t know if RTS/CTS was successful, so…
  – reduced to trying at random times
  – For example, if C tries to send while B is sending, its back-off timer can increase
  – In the long run, B can shut off C completely

• Fix:
  – carrier sense
  – or a DS packet (no carrier sense hardware)
Fairness II

- DS packet: basically a duplicate of the CTS for the benefit of exposed terminals
- If C receives B’s DS, it knows that B’s RTS was acknowledged by A and it knows how long it must wait before attempting to transmit to D
Fairness III

- Issue: A and D are contending but they cannot hear each other
- The one who wins the first contention is more likely to win the second, even more likely to win the third, etc.
- Problem: while B is receiving, C can hear D’s RTS packets but cannot reply to avoid disturbing B
- Solution: have C contend on behalf of D by sending RRTS packet
- Even more fairness issues in the paper!
Fairness - IV

• The proposed system is designed to provide each node with a fair share of bandwidth
• What if certain nodes generate more streams than others?
  • E.g. base stations

Solution: B generates two wait times based on the same backoff counter, one for P1 and one for P2
• It then picks the destination with smallest waiting time
• Small unfairness due to the fact that B->P1 and B->P2 streams do not really contend with each other
IEEE 802.11

- Standard for wireless communication
- MAC-layer uses many of the ideas discussed
  - Basic MAC is a CSMA/CA
    - Carrier-sense and transmit
    - With ACK
  - RTS/CTS exchange is optional
- Allows two modes
  - Ad-hoc
  - Infrastructure
802.11 Details

- much more complex than MACAW (because it’s real, and because it’s designed by committee)
- doesn’t include all of MACAW (less emphasis on fairness, ex. no shared backoff)

- In PCF (base station mode), quite different:
  - Base station polls nodes to see if they have traffic to send (can arbitrate transmissions)
- In DCF (ad-hoc mode)
  - CSMA/CA with ACK
  - Optional RTS/CTS
  - MILD backoff
  - No DS, RRTS etc.
Ad hoc routing

Ad Hoc Routing

• Create multi-hop connectivity among set of wireless, possibly moving, nodes
• Mobile, wireless hosts act as forwarding nodes as well as end systems
• Need routing protocol to find multi-hop paths
  – Needs to be dynamic to adapt to new routes, movement
  – Interesting challenges related to interference and power limitations
Ad-Hoc Routing Requirements

• Distribution paths
  – Multi-hop paths
  – loop-free
  – minimal data transmission overhead
  – multicast?

• Self-starting and adaptive to dynamic topology

• Low consumption of memory, BW, power
  – scalable with numbers of nodes
  – localized effects of link failure
Problems with traditional approaches

• Periodic routing or LS updates require power of sender and of listening receivers
• Topology very dynamic so protocols must converge quickly to avoid black holes
• Not studied in the context of realistic radio propagation models, MAC layers and mobility patterns
Problems using DV or LS

- DV protocols may form loops
  - very wasteful in wireless: bandwidth, power
  - loop avoidance sometimes complex

- LS protocols: high storage and communication overhead

- More links in wireless (e.g., clusters) - may be redundant -> higher protocol overhead
Problems

• Periodic updates waste power
  – tx sends portion of battery power into air
  – reception requires less power, but periodic updates prevent mobile from “sleeping”

• Convergence may be slower in conventional networks but must be fast in ad-hoc networks and be done without frequent updates
Proposed Protocols

- **Destination-Sequenced Distance Vector (DSDV)**
  - hhb, DV protocol w/periodic routing update broadcasts

- **Temporally-Ordered Routing Algorithm (TORA)**
  - on demand creation of hhb routes based on link-reversal

- **Dynamic Source Routing (DSR)**
  - on demand source route discovery

- **Ad Hoc On-Demand Distance Vector (AODV)**
  - combination of DSR and DSDV: on demand route discovery with hop-by-hop routing
DSR

• Components:
  – route discovery
  – route maintenance

• Route discovery - basic idea
  – S broadcasts route-request to D
  – each node forwards request by adding own address and re-broadcasting
  – requests propagate outward until target is found
Route Setup and Maintenance

• A request is forwarded if:
  – node is not the destination
    • If it is, it sends a route reply
  – node not already listed in recorded source route
  – node has not seen request with same sequence number

• Destination \( D \) copies route into a Route-reply packet and sends it back to \( S \)
  – Use source route from cache
  – Reverse learned source route

• Failure detection and recovery
  – From link level notifications
Route Cache

• All source routes learned by a node are kept in route cache
  – reduces cost of route discovery
• If intermediate node receives RR for D and has entry for D in route cache, it responds to RR and does not propagate RR further
  – Need to do this carefully, as this can cause congestion with your neighbors
• Nodes overhearing RR/RP may insert routes in cache
  – Can use information from data packets transiting the node
  – Can promiscuously listen to neighbor’s transmissions
• Scope limit on route requests for reducing discovery overhead
Other Optimizations

• Piggybacking
  – Data messages on the initial route request
  – Reply messages on the reverse route request
  – Need to do this carefully
    • … interacts with route cache optimization

• Hop short-cuts
  – If a node notices that the packet has skipped a hop, it can send an unsolicited route reply

• Optimized error handling
  – Snooping error messages
  – Sender sending a copy of error packet along original path to avoid path asymmetry
Sending Data

- Check cache for route to D
- If route exists then
  - if reachable in one hop
    - send packet
  - else insert routing header to D and send
- If route does not exist, buffer packet and initiate route discovery
Performance Evaluation

• Models for
  – traffic: random pairs sending pseudo-CBR
  – mobility: random waypoint
  – node placement: random

• Metrics
  – path-length relative to optimal
  – message count relative to optimal
Discussion

• Source routing is good for on demand routes instead of a priori distribution
• Route discovery protocol used to obtain routes on demand
  – Caching used to minimize use of discovery
• Periodic messages avoided
• But need to buffer packets