MACAW Bharghavan, Demers, Shenker, Zhang [Bhargahavan94a]

(got to slide 15 on March 2)

CSci551: Computer Networks
SP2006 Thursday Section

John Heidemann

Key ideas

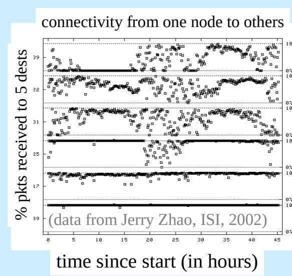
- macaw: Multiple Access Collision Avoidance Wireless
- MAC protocol
  - how to send on the shared channel
- hidden terminal/exposed terminal
  - need to be careful collisions at receiver (senders can't hear each other)
- contention based vs. schedules
  - implications: need to deal with collisions
  - need to be careful about fairness

Wireless MAC Options

- Contention-based vs. token-based/scheduled
  - why contention? statistically better utilization
  - why token? better fairness and gurantees
  - MACAW: contention; 802.11 contention based (ad hoc mode), also scheduled in managed mode
- Base-station vs. ad hoc
  - why base-station? clear who to talk to, how to get to the Internet; use hierarchy; pre-arranged coverage
  - why ad hoc? work without infrastructure; take advantage of relaying
  - MACAW base stations; 802.11 both, but mostly in base-station
    - except that recent Macs seem to support ad hoc mode

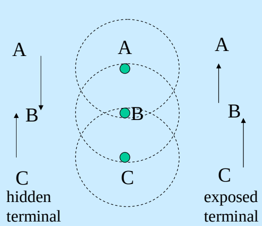
Radio Propagation

- Simple model: fixed tx range
  - propagation can be r^{-3} or r^{-1} (near or far)
  - issues: collisions, capture, interference
  - good simple model, but only an approximation
- Reality is *much* worse
  - multipath fading
  - time-varying effects



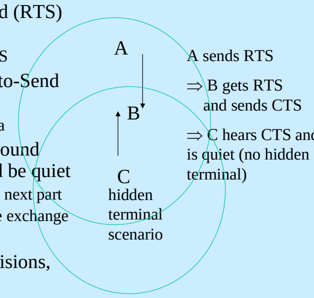
Carrier Sense in Wireless

- Carrier Sense: before transmitting, check if carrier present
  - works in Ethernet
  - why not for wireless? because receiver and sender have different "carrier senses"
- Issues: hidden and exposed terminals



Karn/MACA's RTS-CTS

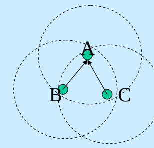
- Src sends Ready-to-Send (RTS) before data
  - overhearers defer for CTS
- Dest replies with Clear-to-Send (CTS)
  - overhearers defer for data
- RTS around src, CTS around dest, so *everyone* should be quiet
  - in MACA: only quiet for next part
  - in 802.11 quite for whole exchange
  - why diff? xxx
- Must also deal with collisions, etc.



- Dx A->B Cx
- D->A->B Cx
- C needs to be quite to not interfere with A, but why D?
- D needs to be quiet because comm is 2-way
  - because B will send an ACK to A, A needs to be able to hear that

Backoff Issues

- Backoff algorithm:
  - backoff counter *bo* estimates population
  - randomly wait [0,*bo*] before sending
  - orignal: binary exponential:
    - *bo* = 0 after success
    - *bo* *= 2 after carrier sense
- Problem: channel capture
  - if I succeed, my *bo* = 0, so I am likely to win again
  - others who fail get slower and slower…

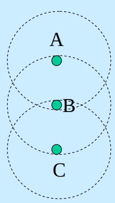


- Fixs:
  - share *bo* (send in each pkt)
  - increase multiplicatively, decrease additively (“MILD”)
  - per-destination backoff

why is backoff hard? (1) need to estimate (changing?) population of senders, (2) need to do this distributedly
wireless-specific problem: *different* estimates of population due to different views of the network

Adding Link-level ACKs

- Wireless losses possible
  - noise or collisions
  - end-to-end argument?
- Add link-level ACK of DATA
  - lost DATA => no ACK => retx
  - lost ACK => sender retx RTS, receiver sends ACK instead of CTS



A sends RTS

⇒ B gets RTS and sends CTS

⇒A sends DATA

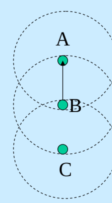
⇒B sends ACK

⇒if no ACK, A resends RTS

end-to-end issues: should we do link-level ACKs? yes, perhaps, if it’s a performance enhancement, but *must* be careful about recovery timescale

Continuing Fairness Problems

- An exposed terminal may not be able to compete effectively
  - C doesn’t know if RTS/CTS was successful,
  - … so reduced to trying at random times
  - tends to back-off more and more
- Fix:
  - carrier sense
  - …or a DS packet
- Doesn’t solve all fairness issues (try A sends to B)



B sends RTS

⇒ A gets RTS and sends CTS; but C misses CTS

⇒ B sends DS

⇒ C hears DS (and data length) and so knows when to try RTS again

⇒ B sends DATA

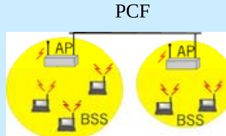
⇒ C knows to RTS after data

Commercializing MACAW: 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, ACK
  - RTS/CTS exchange is optional
- Allows two modes
  - ad-hoc (DCF; Distributed Coordination Function)
  - base-station (PCF; Point Coordination Function)

DCF

PCF



802.11 Comments

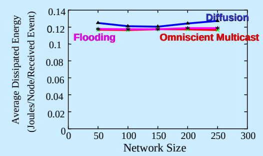
- much more complex than MACAW (because it’s real, and because it’s designed by committee)
- doesn’t include all of MACAW (less emphasis on fairness, ex. no shared backoff)
- spec assumes that all nodes can hear all others at all times
  - but in practice it’s used ad hoc/multi-hop
- infrastructure mode (PCF) is run by access point in a TDMA-like style
- includes several PHY schemes (IR, FH, DS, multiple bit rates)

Energy Use at the MAC

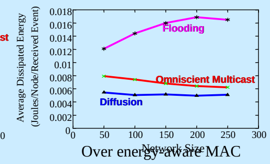
- energy cost for long-range radios
  - listen (idling):receive:send
  - 1:2:10 (exact ratios change depending on radios)
- energy cost for short-range radios
  - listen:receive:send
  - 1:1.1:1.2-2 relative expense of sending is much lower
- MAC time spent?
  - heavy traffic: lots of sending and receiving
  - light traffic: listening

Energy Efficiency in MAC

- Major sources of energy waste
  - Idle listening
    - Energy consumption of some WLAN cards
      idle:receive — 1:1.05 to 1:2
    - Example: *directed diffusion* (Intanagonwiwat 2000)



Over 802.11-like MAC

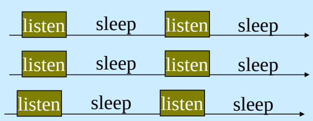


Over energy-aware MAC

(lisent cost is reduced)

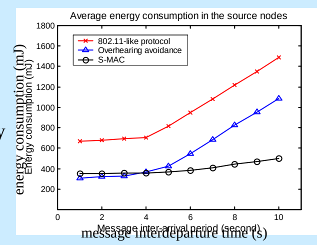
Scheduled MAC Protocols

- Example: S-MAC (Sensor-Media Access Control)
- Approach: put radios to sleep most of the time
  - but they must coordinate so they can still communicate
  - and some will pick conflicting schedules
- details: [Wei Ye, John Heidemann, Deborah Estrin; Infocom 2002]



S-MAC Energy Costs

- at high loads, all protocols are similar
  - all are always busy
  - overhearing avoidance (sleep when others are talking) provides constant savings
- at low loads, S-MAC very helpful
  - costs of listening when nothing to send dominates energy
  - S-MAC's sleeping avoids this



energy consumption (mJ)

message interdeparture time (s)

(heavy tfc) (light traffic)

Other questions/observations?

- radios as broadcast
  - (if you assume omnidirectional antenna)
- some alternatives
  - directional antennae
  - sleep scheduling: sleep when they send traffic not to you (overhearing avoidance)
  - use multi-channel networks (ex. frequencies)
  - or exploit the broadcast information
    - ex: passive acknowledgements
    - or other ways
- what about losses and TCP?
  - problem: TCP thinks loss is due to congestion (not corruption)
  - therefore it slows down
  - what if loss is corruption? resend and not slow down
  - what should we do? either (1) change link layer, or (2) change TCP
    - (1) ex: ARQ, retransmit at the link layer (but be careful how this affects TCP)
- what about variable bitrate?