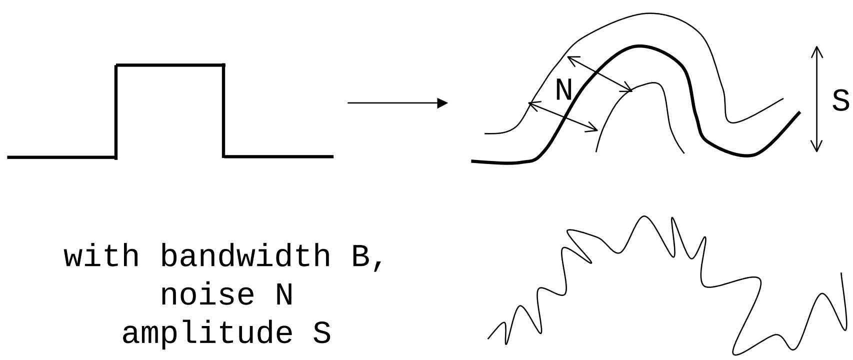
CSE/EE 461 Lecture 3
Link Layer and Media Access

Tom Anderson
tom@cs.washington.edu

Fundamental limits on signal rate



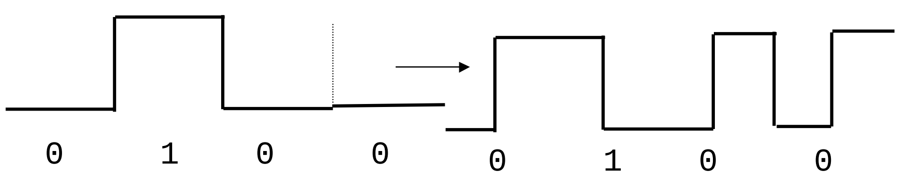
Shannon, Nyquist limit: 2B log (S/2N) bits/second

Clock Recovery

- How does the receiver know when to sample?
  - assume some a priori agreement on rate
  - 1. get in phase: start/stop bit sequence
  - 2. stay in phase despite clock drift
    - precisely identical clocks
    - keep packets shorter than clock drift
    - why not?

Manchester Encoding

- Embed clock into signal
  - Rising = 0; falling = 1
  - Phase-locked loop to stay in sync
    - speed up/slow down clock (slightly!) based on predicted vs. actual transition



4-5 Codes

- Code 4 bits of data into 5 bits of data and clock
  - every bit pattern with a transition
  - ex: 0000 -> 11110; 1111 -> 11101
- Resync every 5 bits

Framing

- Frame is a complete link layer packet
  - need to be able to synchronize sender/receiver on start and end of packet, despite potential bit errors
- Sentinels: special control code that marks start of frame
  - sentinel + fixed sized frames (SONET)
  - sentinel + frame length
  - sentinel + bit stuffing (PPP)
  - physical layer sentinel (ex: unused 4/5 code)

Point-to-Point Protocol (PPP)

- IETF standard, used for dialup and leased lines

Flag 0x7E	(header)	Payload (variable)	(trailer)	Flag 0x7E

- Flag is special and indicates start/end of frame
- Occurrences of flag inside payload must be "stuffed"
  - Replace 0x7E with 0x7D, 0x5E
  - Replace 0x7D with 0x7D, 0x5D

Error Detection/Correction

- Noise, clock sync can flip some of the bits we receive
  - Especially as we approach Shannon/Nyquist limit
- Add redundant data
  - To detect errors to allow retransmission (ARQ = automatic repeat request)
  - To repair errors (FEC = forward error correction)
- Which is more important?

Hamming Distance

- Hamming distance is smallest number of bit differences that turn any one codeword into another
  - ex: code 000 for 0, 111 for 1, Hamming distance is 3
- For code with distance 2d+1:
  - any 2d errors can be detected
    - 001, 010, 110, 101, 011
  - any d errors can be corrected
    - 001 → 000

Algorithms

- Send two complete copies of the packet. Differences imply errors.
  - Can we catch more kinds of errors with less overhead?
- Parity = XOR of all other bits
  - Ex: 0110010 -> 01100101
  - Will detect any single bit error
  - What will it miss?

2D Parity

- Add parity row/column to array of bits
- Detects all 1, 2, 3 bit errors, and many errors with >3 bits.
- Corrects all 1 bit errors

```
0101001  1
1101001  0
1011110  1
0001110  1
0110100  1
1011111  0

1111011  0
```

Checksum

- Sum all the words in the packet and append
  - TCP/IP checksum: interpret the packet as 16 bit words; sum using ones complement; add carry to lsb; invert

```
0101001
1101001
1011110
0001110
0110100
1011111
------------
0010100
```

- What errors can checksums detect? miss?

CRC (Cyclic Redundancy Check)

- Algorithm: Given n bits of data, generate a k bit check sequence that gives a combined n + k bits that are divisible by a chosen divisor C(x)
- Based on mathematics of finite fields
  - "numbers" correspond to polynomials, use modulo arithmetic
  - e.g, interpret 10011010 as $x^7 + x^4 + x^3 + x^1$

How is C(x) Chosen?

- Mathematical properties:
  - All 1-bit errors if non-zero x^k and x^0 terms
  - All 2-bit errors if C(x) has a factor with at least three terms
  - Any odd number of errors if C(x) has (x + 1) as a factor
  - Any burst error < k bits
- There are standardized polynomials known to catch many errors
  - Ethernet CRC-32: 100000100110000010001110110110111

Media Access: Aloha to Ethernet to wireless

How do multiple parties share access to a communication channel (wire or wireless)?

- Delivery: when packet is broadcast, how does the receiver know intended destination?
  - put destination address in frame header
    - – ex: globally unique Ethernet MAC address
  - discard if not intended target
- Arbitration: how do we decide who sends next?

Multiplexing Options

- Frequency division multiple access (FDMA)
  - everyone assigned a different frequency (wavelength)

- Time division multiple access (TDMA)
  - everyone assigned a different time slot

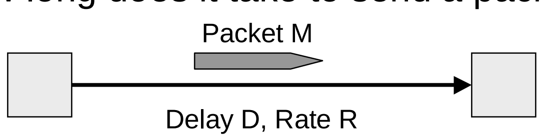
- Code division multiple access (CDMA)
  - multiple senders at a time in each frequency
  - each sender has unique code (1010 vs. 0101 vs. 1100)

Statistical multiplexing

- Assign frequency/time slot on demand
- When is this more efficient than static partitioning? when less efficient?
- How do we arbitrate access?
  - central vs. distributed control?

Digression: Packet Latency

- How long does it take to send a packet?



- Two terms:
  - Propagation delay = distance / speed of light in media
  - Transmission delay = message (bits) / rate (bps)
    - – packet takes up space on wire!
- Slow links stretch bits out in time/space

One-way latency examples

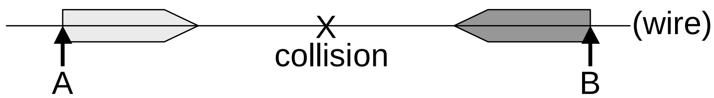
- Either a slow link or long wire makes for large latency
- Dialup with a modem:
  - D = 10ms (say), R = 56Kbps, M = 1000 bytes
  - Latency = 10ms + (1024 x 8)/(56 x 1024) sec = 153ms!
- Cross-country with T3 (45Mbps) line:
  - D = 50ms, R = 45Mbps, M = 1000 bytes
  - Latency = 50ms + (1024 x 8) / (45 x 1000000) sec = 50ms!

ALOHA

- Packet radio network in Hawaii, 1970s
- Wanted distributed allocation
  - no special channels or single point of failure
- Aloha protocol:
  - Just send when you have data!
  - There will be some collisions of course …
  - Throw away garbled frames at receiver (using CRC); sender will time out and retransmit
- Simple, decentralized and works well for low load

Carrier Sense Multiple Access

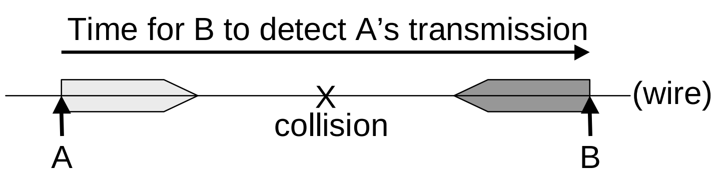
- We can do better by listening before we send (CSMA)
  - good defense against collisions only if “a” is small



- “a” parameter: number of packets that fit on the wire
  - a = bandwidth * delay / packet size
  - a << 1 for LANs; a >> 1 for satellite

CSMA with Collision Detection

- Even with CSMA there can still be collisions.



- What if we detect collisions and abort? CSMA/CD
  - Requires a minimum frame size (“acquiring the medium”), 2 x max propagation delay
  - B must continue sending (“jam”) until A detects collision

What if the Channel is Busy?

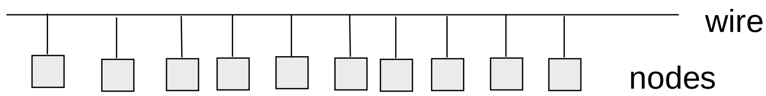
- 1-persistent CSMA
  - Wait until idle then go for it; what happens?
- Non-persistent CSMA
  - Wait a random time and try again
- p-persistent CSMA
  - Wait until idle, then in each time slot, choose to send with prob p
  - What if p is small? What if p is large?

CSMA/CD with Binary Exponential Backoff

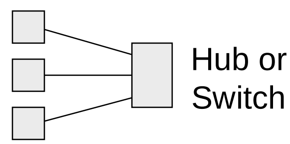
- On collision: jam and exponential backoff
  - Jamming: send bit sequence to ensure collision detection
- Backoff:
  - First collision: wait 0 or 1 frame times at random and retry
  - Second time: wait 0, 1, 2, or 3 frame times
  - Nth time (N<=10): wait 0, 1, …, 2^N-1 times
  - Max wait 1023 frames, give up after 16 attempts
  - Scheme balances average wait with load
    - – what about fairness?

Ethernet

- Ethernet: first practical LAN, Xerox 1980's (802.3)
  - 1-persistent CSMA/CD with binary exponential backoff
  - 10 Mbps over coaxial cable, passive taps
  - Manchester encoding, preamble, 32 bit CRC



- Newer versions
  - Fast (100 Mbps), gigabit (1 Gbps)
  - Switched, point to point wires



Ethernet Frames

Preamble (8)	Source (6)	Dest (6)	Len (2)	Payload (var)	Pad (var)	CRC (4)

- Min frame 64 bytes, max 1500 bytes
- CSMA/CD jam period is 48 bits
- Max length 2.5km, max between stations 500m (repeaters)
- Addresses unique per adaptor; globally assigned
- Broadcast media:
  - ARP, multicast, promiscuous mode monitoring

Ethernet Evaluation

- Fairness -- backoff favors latest arrival
  - max limit to delay
  - no history -- unfairness averages out
- Stable performance under increasing load
  - Much better than Aloha!
  - Works very well in practice
- Source of protocol inefficiency: collisions
  - What happens as bit rates increase?
    - Need to shorten wires and increase frame size

Why Did Ethernet Win?

- Competing technology: token rings (FDDI)
  - “right to send” rotates around ring
  - supports fair, real-time bandwidth allocation
- Failure modes
  - token rings -- network unusable
  - Ethernet -- node detached
- Volume
- Flexibility – Ethernet switches added later

Wireless Communication

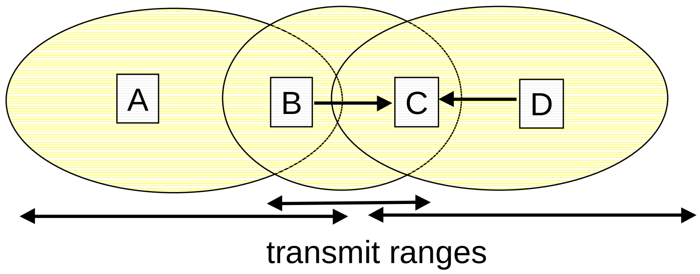
Wireless is more complicated than wired …

- Cannot detect collisions
  - Transmitter swamps co-located receiver
- Different transmitters have different coverage areas
  - Asymmetries lead to hidden/exposed terminal problems

CSMA with Collision Avoidance

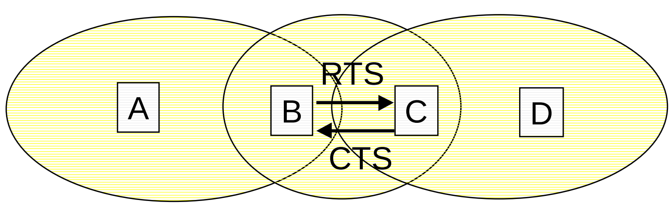
- Since we can’t detect collisions, we avoid them
  - CSMA/CA as opposed to CSMA/CD
- If medium busy, choose random backoff interval
  - Wait for that many idle timeslots to pass before sending
- If a collision is inferred, retransmit with binary exponential backoff (like Ethernet)
  - Use CRC and ACK from receiver to infer “no collision”
  - Again, exponential backoff helps us adapt “p” as needed

Hidden and Exposed Terminals



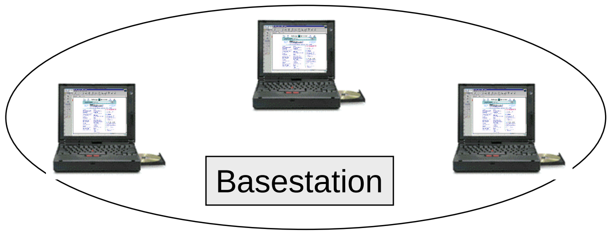
- Hidden terminals: B and D can both send to C but can't hear each other
- Exposed terminals: B, C can hear each other but can safely send to A, D

RTS / CTS Protocols (MACA)



- B asks C: Request To Send (RTS)
- A hears RTS and defers to allow the CTS
- C replies to B with Clear To Send (CTS)
- D hears CTS and defers to allow the data
- B sends to C

802.11 Wireless LANs



- Emerging standard: wireless plus wired system or pure wireless (ad hoc)
- Avoids collisions (CSMA/CA (p-persistence), RTS/CTS
- Built on new links (spread spectrum, or diffuse infrared)