Lecture 26: Ethernet

CSE 123: Computer Networks
Aaron Schulman
Lecture 26 Overview

- Finishing up media access
  - Ethernet
  - Contention-free methods (rings)

- Moving beyond one wire
  - Link technologies have limits on physical distance
  - Also frequently on number of hosts connected
Q: What is max fraction slots successful?
A: Suppose $n$ stations have packets to send

- Each transmits in slot with probability $p$
- Prob[successful transmission], $S$, is:

$$S = p \cdot (1-p)^{n-1}$$

- any of $n$ nodes:

$$S = \text{Prob[one transmits]} = np(1-p)^{n-1}$$

(optimal $p$ as $n \to \infty = 1/n$)

$$= \frac{1}{e} = .37$$

At best: channel used for useful transmissions 37% of time!
Carrier Sense (CSMA)

- Aloha transmits even if another host is transmitting
  - Thus guaranteeing a collision

- Instead, listen first to make sure channel is idle
  - Useful only if channel is frequently idle
  - Why?

- How long to be confident channel is idle?
  - Depends on maximum propagation delay
  - Small (<<1 frame length) for LANs
  - Large (>>1 frame length) for satellites
Retransmission Options

- **non-persistent CSMA**
  - Give up, or send after some random delay
  - Problem: may incur larger delay when channel is idle

- **1-persistent CSMA**
  - Send as soon as channel is idle
  - Problem: blocked senders all try to send at once

- **$P$-persistent CSMA**
  - If idle, send packet with probability $p$; repeat
  - Make sure $(p \times n) < 1$
Even with CSMA there can still be collisions. Why?

- If nodes can detect collisions, abort! *(CSMA/CD)*
  - Requires a minimum frame size ("acquiring the medium")
  - $B$ must continue sending ("jam") until $A$ detects collision

- Requires a **full duplex** channel
  - Wireless is typically half duplex; need an alternative
Collision Detection (CD)

- How can A know that a collision has taken place?
  - Worst case:
    - Latency between nodes A & B is $d$
    - A sends a message at time $t$ and B sends a message at $t + d$ – epsilon (just before receiving A’s message)
  - B knows there is a collision, but not A… A must keep transmitting until it can tell if a collision occurred
  - How long? $2 * d$

- IEEE 802.3 Ethernet specifies max value of $2d$ to be 51.2us
  - This relates to maximum distance of 2500m between hosts
  - At 10Mbps it takes 0.1us to transmit one bit so 512 bits take 51.2us to send
  - So, Ethernet frames must be at least 64B (512 bits) long
    - Padding is used if data is too small

- Send jamming signal to insure all hosts see collision
  - 48 bit signal
Ethernet

- First **local area network** (LAN)
  - Developed in early ’70s by Metcalfe and Boggs at PARC
  - Originally 1Mbps, now supports 10Mbps, 100Mbps, 1Gbps, 10Gbps, 40Gbps, and 100Gbps flavors (400G in dev)

- Currently **the** dominant LAN technology
  - Becoming the dominant WAN technology
**Classic Ethernet**

- IEEE 802.3 standard wired LAN (modified 1-persistent CSMA/CD)
- Classic Ethernet: 10 Mbps over coaxial cable
  - All nodes share same wire
  - Max length 2.5km, max between stations 500m

```
+--------+--------+--------+ (wire) +--------+--------+
| nodes  | nodes  |        | (wire) | nodes  | nodes  |
```

- Framing
  - Preamble, 32-bit CRC, variable length data
  - Unique 48-bit address per host (bcast & multicast addr too)

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

- Problems with random delay with fixed mean
  - Few senders = unnecessary delay
  - Many senders = unnecessary collisions

- Binary exponential back-off balances delay w/load
  - First collision: wait 0 or 1 min frame times at random, retry
  - Second time: wait 0, 1, 2, or 3 times
  - Nth time (n<=10): wait 0, 1, ..., $2^n$-1 times
  - Max wait 1023 frames; give up after 16 attempts
Capture Effect

- Randomized access scheme is not fair

- Suppose stations A and B always have data to send
  - They *will* collide at some time
  - Both pick random number of “slots” (0, 1) to wait
  - Suppose A wins and sends
  - Next time they collide, B’s chance of winning is halved
    » B will select from 0,1,2,3 due to exponential back-off

- A keeps winning: said to have *captured* the channel
Ethernet Performance

- Much better than Aloha or CSMA in practice

- Source of protocol inefficiency: still collisions
  - More efficient to send larger frames
    » Acquire the medium and send lots of data
  - Less efficient if
    » More hosts – more collisions needed to identify single sender
    » Smaller packet sizes – more frequent arbitration
    » Longer links – collisions take longer to observe, more wasted bandwidth
Contention-free Protocols

- Problem with fixed partitioning:
  - Inefficient at low load (idle channels)

- Problem with contention-based protocols:
  - Inefficient at high load (collisions)

- Ideal(?): Contention-free protocols
  - Try to do both by explicitly taking turns
  - Can potentially also offer guaranteed bandwidth, latency, etc.
Contention-free Approaches

Polling

- Master node “invites” slave nodes to transmit in turn
  - Request to Send (RTS), Clear to Send (CTS) messages

- Problems:
  - Polling overhead
  - Latency
  - Single point of failure (master)

Token Passing

- Control **token** passed from one node to next sequentially.

- Problems:
  - Token overhead
  - Latency
  - Single point of failure (token)
Token Ring (802.5)

- Token rotates “permission to send” around nodes
- Sender injects packet into ring and removes later
  - Maximum token holding time (THT) bounds access time
  - Early or delayed token release
  - Round robin service, acknowledgments and priorities
- Monitor nodes ensure health of ring (alerts on failures)
FDDI
(Fiber Distributed Data Interface)

- Roughly a large, fast token ring
  - First real use of fiber optics in a LAN
  - 100 Mbps and 200km (FDDI) vs 4/16 Mbps and local (802.5)
  - Dual counter-rotating rings for redundancy
  - Complex token holding policies for voice etc. traffic

- Token ring advantages
  - No contention, bounded access delay
  - Support fair, reserved, priority access

- Disadvantages
  - Complexity, reliability, scalability
Why Did Ethernet Win?

- Failure modes
  - Token rings – network unusable
  - Ethernet – node detached

- Good performance in common case

- Completely distributed, easy to maintain/administer

- Easy incremental deployment

- Volume → lower cost → higher volume ….
Summary of Media Access

- How to divide shared channel among different users
  - Fixed partitioning (FDMA, TDMA, CDMA)
    - Guaranteed bandwidth for each user, but wasteful when not used and can’t allocate different bandwidth to different users
  - Contention-based protocols (CSMA, CSMA/CD)
    - Try and backoff if fail; dynamic allocation of bandwidth on demand, works well at load load but collisions a problem at high load
  - Contention-free protocols (Token Ring, FDDI, RTS/CTS)
    - Explicit turn-taking; strong guarantees on access time and can make bandwidth guarantees, but complex and fragile to failure

- But… aren’t there limits to what we can do with one shared channel?