Computer Networks

Medium Access Sublayer (Part I)
Topics

- Introduction
- Multiple Access Protocols
- Ethernet
- Wireless LAN Protocols
- Bridges
- Misc (brief)
  - High-Speed LANs
  - Satellite Networks
Introduction

- Remember, two categories of networks
  - point-to-point
  - broadcast

- Key issue is who gets channel
  - example: 6-person conference call

- Many protocols to decide

- Medium Access Control sublayer
  - lower part of data-link layer, but easier here

- Many LANs multiaccess
  - satellites, too
Fixed Channel Allocation

- Static channel allocation
  - FDM, TDM
FDM

- Time delay $T$
- Capacity $C$ bps
- Arrival rate $\lambda$ frames/sec
- Frames mean $1/\mu$ bits

- Divide into $N$ channels
- Each channel $C/N$ bps

TDM is the same

$$T = \frac{1}{\mu C - \lambda}$$

$$= \frac{1}{\mu(C/N) - (\lambda/N)}$$

$$= \frac{N}{\mu C - \lambda}$$

$$= NT$$
Dynamic Channel Allocation in LANs and MANs: Assumptions

- Station Model
  - $N$ independent stations

- Single Channel Assumption.
  - One shared channel for transmission

- Collision Assumption.
  - Garbled if transmissions overlap

- (a) Continuous Time.
  - (b) Slotted Time.

- (a) Carrier Sense.
  - (b) No Carrier Sense.
Multiple Access Protocols

- ALOHA
- Carrier Sense Multiple Access Protocols
- Collision-Free Protocols
- Limited-Contention Protocols
- Wireless LAN Protocols
ALOHA - A Family of Contention Protocols

- 1970’s, Abramson
- University of Hawaii
- Ground based broadcasting, packet radio
  - generalizes to uncoordinated users competing for single, shared channel
- Pure ALOHA
  - no time slots
- Slotted ALOHA
  - time slots for frames
# Pure ALOHA

- Transmit whenever you want
- Detect collisions after sending
  - checksum error
- If collision, wait random time and retry

<table>
<thead>
<tr>
<th>User</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
</tr>
</tbody>
</table>
Pure ALOHA == Pure Chaos?

Assume infinite collection of stations

Users in two states: typing or waiting

User typing a line. When done, transmit it.
  - user waiting for response. When done, typing.

Frame time is time to put frame on wire
  - frame length / bit rate (fixed frame length)

Mean number of new frames per frame time
  - $N$
  - What does $N > 1$ mean?
Analysis of Pure ALOHA

- Stations also re-generate collided frames
  - $G$ is old plus new frames
  - $G > N$? $G = N$? $G < N$?
- Low load ($N \approx 0$), few collisions: $G \approx N$
- High load, many collisions: $G > N$
- Throughput per frame time is $G$ times probability of frame having zero collisions:

  $$S = G P_0$$

  - ex: $G = .5$, $P_0 = .5$ so $S = .25$
  - Note: $P_0$ is probability of successful transmission
Frame Collisions

Collides with the start of the shaded frame

Collides with the end of the shaded frame

Vulnerable

$\text{Collides with the start of the shaded frame}$

$\text{Collides with the end of the shaded frame}$

$\text{Vulnerable}$

$\text{Time}$
Analysis of Pure ALOHA (cont.)

- Probability $k$ frames generated per frame time
  \[ G^k e^{-G} \]
  \[ \Pr[k] = \frac{G^k e^{-G}}{k!} \]
  \[ \Pr[0] = e^{-G} \]

- Need two frame times empty, $2G$ generated
  - for two slots, \[ \Pr[0] = e^{-2G} \]

- Using $S = G \Pr[0]$, throughput per frame time
  \[ S = Ge^{-2G} \]
Pure ALOHA
Offered Load vs. Throughput

Max at \( G = 0.5, S = \frac{1}{2}e \), only about 0.184 (18%)!

– Can we do better?
Slotted ALOHA

- Divide time into intervals, one for each frame
- Stations agree upon time intervals
  - one can “pip” as time keeper, like a clock
- Users transmit only at beginning of slot
- Need one frame time to be empty, $G$ generated
  - for one slot, $\text{Pr}[0] = e^{-G}$
- Throughput
  $$S = Ge^{-G}$$
Slotted ALOHA
Offered Load vs. Throughput

Max at $G = 1, S = 1/e$, only about 0.368 (37%)
– This is not Ethernet!
Last Thoughts on Slotted ALOHA

- Best (G = 1):
  - 37% empty
  - 37% success
  - 26% collisions

- Raising G, reduces empties but increases collisions exponentially

- Expected transmissions (includes original)
  \[ E = e^G \]
  - G=0, then 1 transmission; G=1 then 2.X trans.

- Small increase in load, big decrease in perf
Carrier Sense Multiple Access - CSMA Protocols

- Sending without paying attention is obviously limiting
- In LANs, can detect what others are doing
- Stations listen for a transmission
  - carrier sense protocols
Persistent and Nonpersistent

\[ \text{\textit{l-persistent} CSMA} \]
- detect, send at first chance
- wait if another sending
- longer delay, more collisions

\[ \text{\textit{non-persistent} CSMA} \]
- if empty, send
- if not, less greedy, waits random time then repeats
- fewer collisions, longer delay

\[ \text{\textit{p-persistent} CSMA} \]
- if empty, sends with probability \( p \)
- defers with probability \( q = 1 - p \)
Carrier Sense Multiple Access
CSMA with Collision Detection

- If detect collision, stop transmitting
  - frame will be garbled anyway
- CSMA with Collision Detection (CD)
CSMA/CD Closing Comments

- How long until realize a collision? Time to travel length of cable? Why not?
- Propogation $\tau$, need $2\tau$ to “seize” the line
- Model $2\tau$ slot as slotted ALOHA
- 1-km cable has $\tau \approx 5$ $\mu$sec
- Collision detection *analog*
  - special hardware encoding so can detect
- Does not guarantee reliable delivery
- Basis IEEE 802.3 (*Ethernet*)
Collision-Free Protocols

- Collisions still occur in CSMA/CD
- More so when “wire” long (large $\tau$)
- Short frames, too, since contention period becomes more significant
- Want collision free protocols
- Need to assume $N$ stations have numbers 0 to (N-1) wired in
Bit-Map Protocol

- Have N contention slots
- Station N puts 1 in slot N-1, else 0
  - ex: station 0 wants to send, 1 in 0th slot
Bit-Map Protocol Performance

- $N$ contention slots, so $N$ bits overhead /frame
- $d$ data bits
- Station wants to transmit, waits
  - Low numbered: avg $N/2$ slots (current) + $N$ for next
  - High numbered: avg. $N/2$
  - Combined avg. delay: $N$
- Efficiency under low load (1 sending):
  - $d/(N+d)$
  - average delay: $N/2$
- High load (N sending): can prorate overhead
  - $d/(d+1)$
  - average delay: $N(d+1)/2$
Where the Heck Were We?

- Introduction
- Multiple Access Protocols
  - contention
  - collision-free
- Ethernet
- Wireless LAN Protocols
- Bridges
- Misc (brief)
  - High-Speed LAN
Binary Countdown

- Instead of 1 bit per station, encode in binary
  - transmit address in binary
- Assume all stations see inserted bits instantaneously
- When multiple transmit, OR together
- When a station sees high-order 1 bit where it has a zero, it gives up
Binary Countdown Performance

- Efficiency: \( \frac{d}{d + \log_2 N} \)
- Sender address as first field and \textit{no} overhead
- Fairness/Unfairness?
  - Mok and Ward (1979): Use virtual station numbers
  - \( C, H, D, A, G, B, E, F \) are 7, 6, 5, 4, 3, 2, 1, 0
  - \( D \) sends: \( C, H, A, G, B, E, F, D \)
Contestion vs. Collision-Free

- Contention better under low load. *Why?*
- Collision-free better under high load. *Why?*
- Hybrid: *limited contention protocols*
- Instead of symmetric contention, asymmetric
- Divide into groups. Each group contents for same slot.
- How to assign to slots?
  - 1 per slot, then collision free (Binary Countdown)
  - All in same slot, then contention (CSMA/CD)
Adaptive Tree Walk Protocol

- U.S. Army test for Syphilis
  - Test group, if negative all ok
  - If positive, then split in two and re-test
Adaptive Tree Walk Protocol

- Where to begin searching (entire army?)
  - if heavily loaded, not at the top since there will always be a collision
- Number levels 0, 1, 2 …
- At level $i$, $1/2^i$ stations below it
  - ex: level 0, all stations below it, 1 has 1/2 below…
- If $q$ stations want to transmit, then $q/2^i$ below
- Want number below to be 1 (no collisions)
  - $q/2^i = 1$, $i = \log_2 q$
Other Improvements

If collision at 1, 2 idle, do we need to search 3?
Heck, Here We Are

- Introduction
- Multiple Access Protocols
  - contention
  - collision-free
- Ethernet
- Wireless LAN Protocols
- Bridges
- Misc (brief)
  - High-Speed LANs
  - Satellite Networks
Ethernet

- Ethernet Cabling
- Manchester Encoding
- The Ethernet MAC Sublayer Protocol
- The Binary Exponential Backoff Algorithm
- Ethernet Performance
- Switched Ethernet
- Fast Ethernet
- Gigabit Ethernet
- IEEE 802.2: Logical Link Control
Ethernet (IEEE 802.3)

- Began as ALOHA, added carrier sense
- Xerox PARC built 3 Mbps version for workstations and called it *Ethernet*
  - old scientist dudes thought waves propagated through substance called “ether”, so a geeky joke
- Xerox, DEC and Intel made 10 Mbps standard
  - 1 to 10 Mbps
  - not “Ethernet”, but close enough
Ethernet Cabling

- **10Base5 - “Thick Ethernet”**
  - 10 Mbps, 500 meters

- **10Base2 - “Thin Ethernet” or “Thinnet”**
  - BNC connectors, or T-junctions
  - Easier and more reliable than 10Base5
  - But only 200 meters and 30 stations per segment

- All on one line, then difficult to find break
  - *domain reflectometry*
  - *hubs*

- **10BaseT (Twisted pair)**

- **10BaseF (Fiber)**
## Kinds of Ethernet Cabling

<table>
<thead>
<tr>
<th>Name</th>
<th>Cable</th>
<th>Max. seg.</th>
<th>Nodes/seg.</th>
<th>Advantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>10Base5</td>
<td>Thick coax</td>
<td>500 m</td>
<td>100</td>
<td>Original cable; now obsolete</td>
</tr>
<tr>
<td>10Base2</td>
<td>Thin coax</td>
<td>185 m</td>
<td>30</td>
<td>No hub needed</td>
</tr>
<tr>
<td>10Base-T</td>
<td>Twisted pair</td>
<td>100 m</td>
<td>1024</td>
<td>Cheapest system</td>
</tr>
<tr>
<td>10Base-F</td>
<td>Fiber optics</td>
<td>2000 m</td>
<td>1024</td>
<td>Best between buildings</td>
</tr>
</tbody>
</table>

Three kinds of Ethernet cabling.

(a) 10Base5, (b) 10Base2, (c) 10Base-T.
Cable Topologies

Cable topologies. (a) Linear, (b) Spine, (c) Tree, (d) Segmented.
Encoding

- 0 volts for 0 and 5 volts for 1 can be misleading
- Want start, middle and end of each bit without reference to external clock
  - Manchester Encoding
  - Differential Manchester Encoding uses changes

<table>
<thead>
<tr>
<th>Bit stream</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary encoding</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manchester encoding</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Differential Manchester encoding</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Transition here indicates a 0
Lack of transition here indicates a 1
### Ethernet Protocol

- **Preamble:** 10101010 to allow clock synch
- **Start of Frame:** 10101011
- **Source and Destination addr:** 2 or 6 bytes
  - 1 for high order bit means “multicast”
  - all 1’s means “broadcast”
- **Length:** data length, 46 to 1500
  - very small frames, problems, so pad to 46

<table>
<thead>
<tr>
<th>Bytes</th>
<th>8</th>
<th>6</th>
<th>6</th>
<th>2</th>
<th>0-1500</th>
<th>0-46</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>Preamble</td>
<td>Destination address</td>
<td>Source address</td>
<td>Type</td>
<td>Data</td>
<td>Pad</td>
<td>Check-sum</td>
</tr>
<tr>
<td>(b)</td>
<td>Preamble</td>
<td>SOF</td>
<td>Destination address</td>
<td>Source address</td>
<td>Length</td>
<td>Data</td>
<td>Pad</td>
</tr>
</tbody>
</table>

Frame formats. (a) DIX Ethernet, (b) IEEE 802.3.
Frame must be $> 2\tau$

Otherwise, how to tell collision from short frame?
Collision Action?

- Each slot of length $2\tau$
- If collision, then wait 0 or 1 slot
- If another collision, then wait 0, 1, 2, 3 slots
- If another collision, then wait 0 to $2^3-1$ slots
- After $i$ collisions, wait 0 to $2^i-1$ slots
  - called *binary exponential backoff*
  - why is this a good idea? Consider other options
- After 10 collisions, wait 0 to 1023 slots
- After 16 collisions, throw in the towel
Now, Where Were We?

- Introduction
- Multiple Access Protocols
- IEEE 802 Standard
  - Ethernet (802.3)
- Wireless LAN Protocols
- Misc