Lecture 8:
Carrier Sense Multiple Access

CSE 123: Computer Networks
Alex C. Snoeren

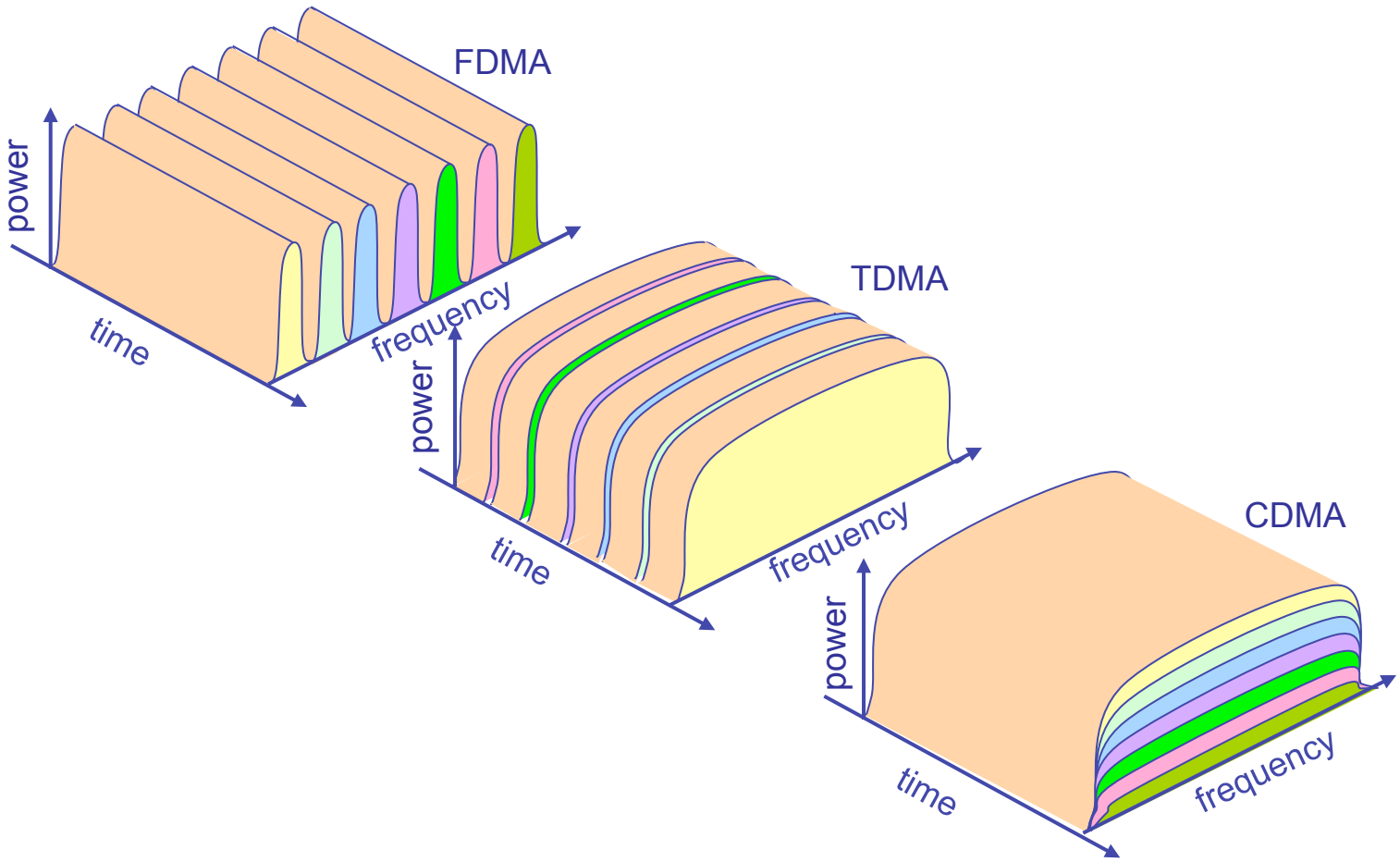
HW 2 due next Wednesday

UCSDCSE

Lecture 8 Overview

- Methods to share physical media: multiple access
  - Random access
- Channelizing mechanisms
- Contention-based mechanisms
  - Aloha
  - Ethernet

Partitioning Visualization



Courtesy Takashi Inoue

Problem w/Channel partitioning

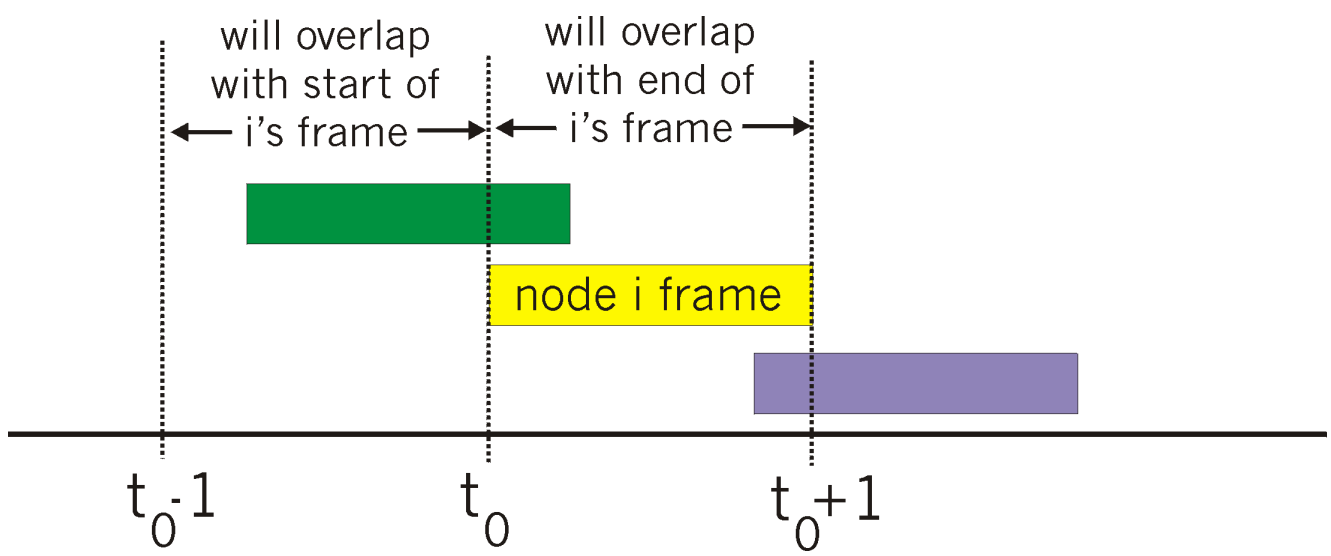
- Not terribly well suited for random access usage
  - Why?
- Instead, design schemes for more common situations
  - Not all nodes want to send all the time
  - Don’t have a fixed number of nodes
- Potentially higher throughput for transmissions
  - Active nodes get full channel bandwidth

Aloha

- Designed in 1970 to support wireless data connectivity
  - Between Hawaiian Islands—rough!
- Goal: distributed access control (no central arbitrator
  - Over a shared *broadcast* channel
- Aloha protocol in a nutshell:
  - When you have data **send it**
  - If data doesn't get through (receiver sends acknowledgement) then **retransmit after a random delay**
  - Why not a fixed delay?

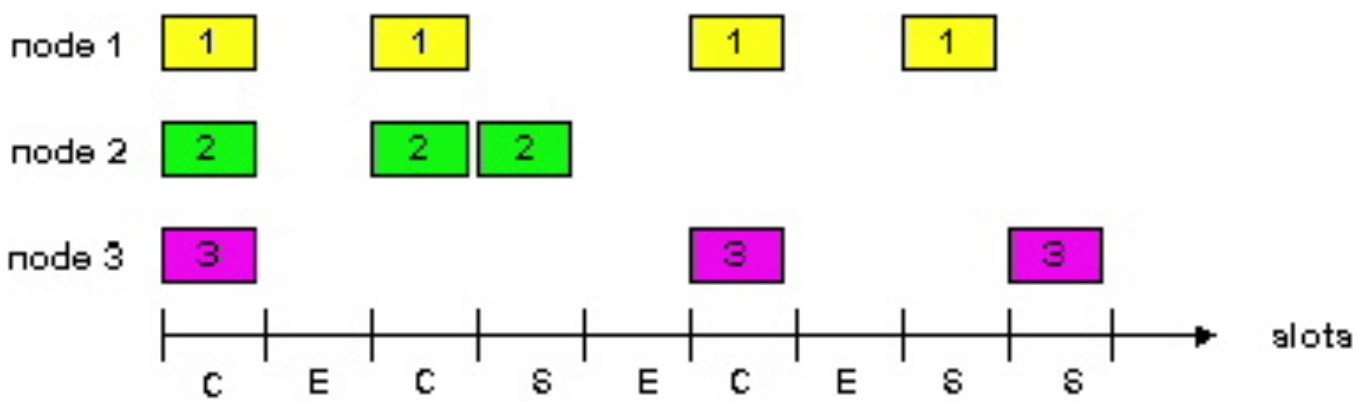
Collisions

- Frame sent at t_0 collides with frames sent in $[t_0-1, t_0+1]$
  - Assuming unit-length frames
  - Ignores propagation delay



Slotted Aloha

- Time is divided into equal size slots (frame size)
- Host wanting to transmit starts at start of next slot
  - Retransmit like w/Aloha, but quantize to nearest next slot
- Requires **time synchronization** between hosts



Success (S), Collision (C), Empty (E) slots

Channel Efficiency

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:

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

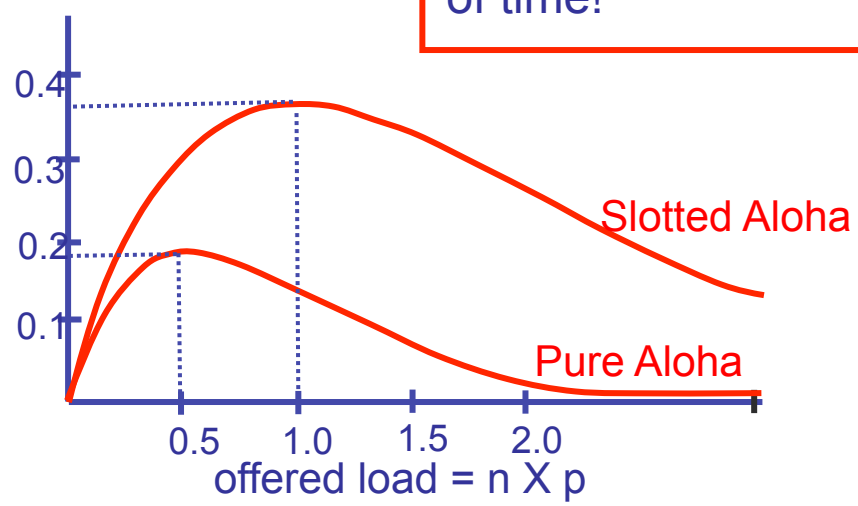
- any of *n* nodes:

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

(optimal *p* as *n*->infinity = *1/n*)

= *1/e* = *.37*

At best: channel used for useful transmissions 37% of time!



Carrier Sense (CSMA)

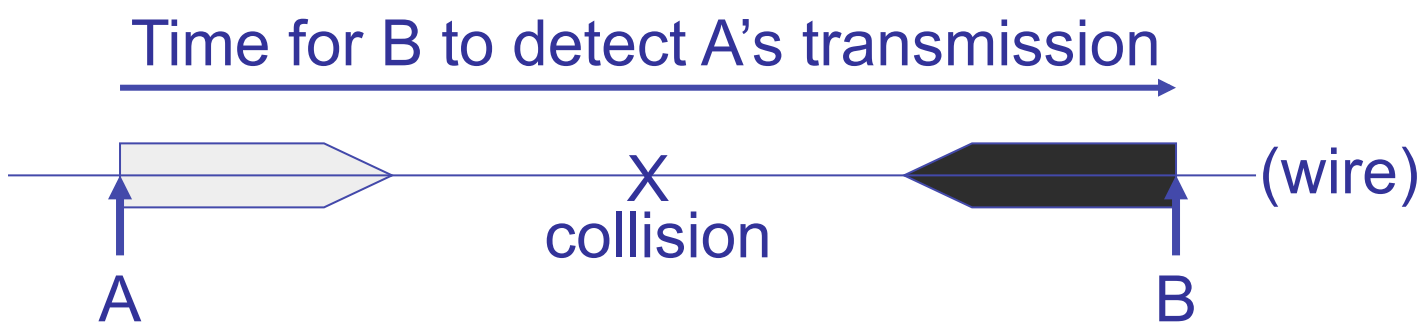
- Aloha transmits even if another host is transmitting
  - Thus guaranteeing a *collision*
- Instead, listen *first* to make sure channel is idle
  - Useful only if channel is frequently idle
  - Why?
- How long to be confident channel is idle?
  - Depends on maximum propagation delay
  - Small (<<1 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 $(p * n) < 1$

Jamming

- Even with CSMA there can still be collisions. Why?



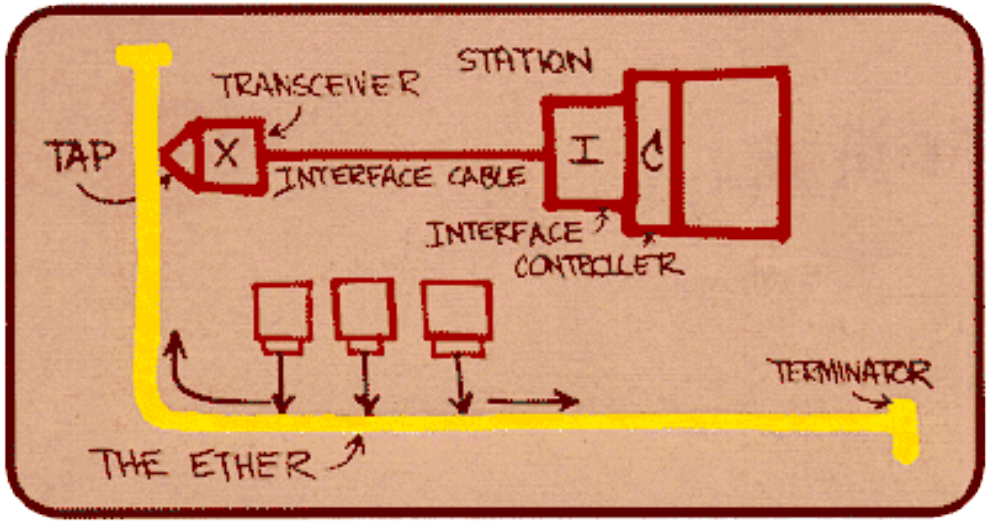
- If nodes can detect collisions, abort! (CSMA/CD)
  - Requires a minimum frame size ("acquiring the medium")
  - *B* must continue sending ("jam") until *A* detects collision
- Requires a full duplex channel
  - Wireless is typically half duplex; need an alternative

Collision Detection

- 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… B must keep transmitting so A knows that its packet has collided
  - 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

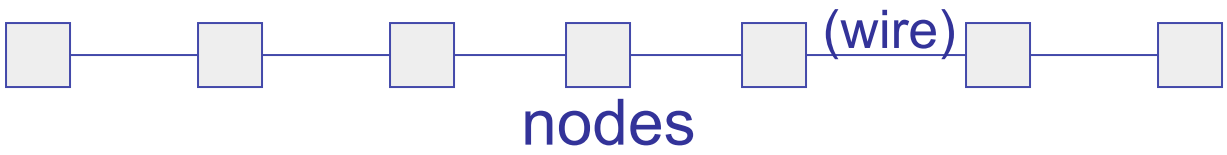
Ethernet

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



- Framing
  - Preamble, 32-bit CRC, variable length data
  - Unique 48-bit address per host (bcast & multicast addrs 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
  - *N*th 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 the 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: 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

For Next Time

- Read 3-3.2 in P&D
- Keep going on the project…
- Really, have a great weekend!