CS 640 Introduction to Computer Networks

Lecture 5

Today's lecture

- Ethernet

Ethernet History

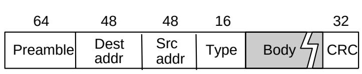
- Developed by Bob Metcalfe and others at Xerox PARC in mid-1970s
- Roots in Aloha packet-radio network
- Standardized by Xerox, DEC & Intel in 1978
- LAN standards define layer 1 and 2
  - IEEE 802.3 (CSMA/CD - Ethernet) standard – originally 2Mbps
  - IEEE 802.3u standard for 100Mbps Ethernet
  - IEEE 802.3z standard for 1,000Mbps Ethernet

Ethernet Overview

- Most popular LAN technology
- Bandwidths: 10Mbps, 100Mbps, 1Gbps
- Max bus length: 2500m
  - 500m segments with 4 repeaters
- Bus and Star topologies are used to connect hosts
  - Hosts attach via Ethernet transceiver or hub or switch
    - Detects line state and sends/receives signals
  - All hosts on an Ethernet compete for access to the medium
    - Switches break this model

Ethernet framing and error detection

- Preamble is a sequence of 7 bytes, each set to "10101010"
  - Used to synchronize receiver before actual data is sent
  - Body can contain up to 1500 bytes of data
- CRC is used to detect errors



Ethernet multiplexing

- Addresses
  - unique, 48-bit unicast address assigned to each adapter
    - Example: **`08:00:20:b1:25:d2`**
    - Each manufacturer gets their own address range
  - broadcast: all **`1`**s
  - multicast: first bit is **`1`**
- Type field is a demultiplexing key used to determine which higher level protocol the frame should be delivered to

Multiple Access Methods

- Centralized master-slave protocols
  - Master decides who sends when
- Fixed assignment
  - Partition channel so each node gets a slice of the bandwidth
  - Inefficient for bursty data
- Distributed token-based
  - Take turns using the channel (Token ring)
- Contention-based
  - Nodes contend equally for bandwidth and recover from collisions (Aloha, Ethernet)

Aloha

- Developed in late 60's by Norm Abramson at Univ. of Hawaii for use with packet radio systems
  - Any station can send data at any time
  - Receiver sends an ACK for data
  - Timeout for ACK signals that there was a collision
    - What happens if timeout is poorly timed?
  - If there is a collision, resend data after random backoff
- Utilization was pretty bad
  - Max utilization = 18%
- Slotted Aloha (send only when slot starts) helped
  - Max utilization = 36%

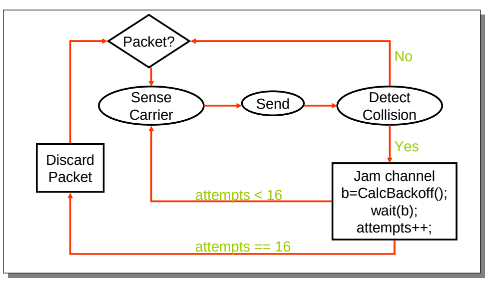
Ethernet Media Access Control

- CSMA/CD
  - CS = carrier sense
    - Send only if medium is idle
  - MA = multiple access
  - CD = collision detection
    - Stop sending immediately if collision is detected

Ethernet's MAC Algorithm

- If line is idle (no carrier sensed)
  - send packet immediately
  - upper bound message size of 1500 bytes
  - must wait 9.6us between back-to-back frames
- If line is busy (carrier sensed)
  - wait until idle and transmit packet immediately
    - called *1-persistent* sending
- If collision detected
  - Stop sending and jam signal
  - Try again later

State Diagram for CSMA/CD

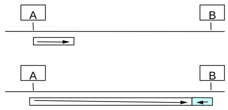


Collisions

Collisions happen when two adaptors transmit at the same time (adaptors sense collision based on voltage differences)

- Both found line to be idle
- Both had been waiting to for a busy line to become idle

A starts at time 0



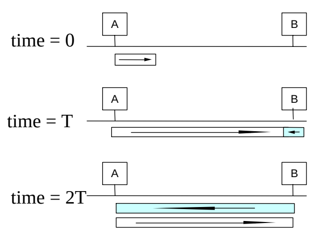
Message almost there at time T when B starts – collision!

How can we be sure A knows about the collision?

Collision Detection

- How can A know that a collision has taken place?
  - There must be a mechanism to ensure retransmission on collision
  - A's message reaches B at time T
  - B's message reaches A at time 2T
  - So, A must still be transmitting at 2T
- IEEE 802.3 specifies max value of 2T 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 (64B) take 51.2us to send
  - So, Ethernet frames must be at least 64B long
    - 14B header, 46B data, 4B CRC
    - Padding is used if data is less than 46B
- Send jamming signal after detecting collision so all hosts see it

Collision Detection contd.



Exponential Backoff

- If a collision is detected, delay and try again
- Delay time is selected using binary exponential backoff
  - 1st time: choose K from {0,1} then delay = K * 51.2us
  - 2nd time: choose K from {0,1,2,3} then delay = K * 51.2us
  - *nth* time: delay = $K \times 51.2$us, for $K=0..2^n - 1$
  - give up after 16 tries and report error
- If delay were not random, could get continual collisions
- Why not just choose from small set for K?
  - This works fine for a small number of hosts
  - Large number of nodes would result in more collisions

MAC Algorithm for Receiver Side

- Senders handle all access control
- Receivers simply read frames with acceptable address
  - Address to host
  - Address to broadcast
  - Address to multicast to which host belongs
  - All frames if host is in promiscuous mode

Experiences with Ethernet

- Ethernets work best under light loads
  - Utilization over 30% is considered heavy
    - Network capacity is wasted by collisions
- Most networks are limited to about 200 hosts
  - Specification allows for up to 1024
- Transport level flow control helps reduce load (number of back to back packets)
- Ethernet is inexpensive, fast and easy to administer!

Why did Ethernet Win?

- There are LOTS of LAN protocols
- **Price**
- Performance
- Availability
- Ease of use
- Scalability