CMPE 150:
Introduction to Computer Networks

Dr. Chane L. Fullmer
chane@cse.ucsc.edu
Class Information

- Class web site: 
  www.soe.ucsc.edu/classes/cmpe150/Spring03/

- Class Newsgroup
  - ucsc.class.cmpe150

- TA Section
  - Monday/Wednesdays @ 5:00PM
    - Baskin *White Boards* area
(Optional) Class Project

- Network programming project
  - In lieu of taking final examination
- Goal:
  - Build an FTP client/server from scratch
  - Using ‘C’ language
- Details on web page... now...
  - Due by June 4th...
Homework Assignments

Homework assignment #1

Due now...... please hand in 😊
CMPE 150: Introduction to Computer Networks

LECTURE 3:

Medium Access Control, Part I
Medium Access Control (MAC) Protocols

- Used to share the use of transmission media that can be accessed concurrently by multiple users.
Functions at The Link Layer

- **MAC:**
  - Framing and synchronization
  - Error checking
  - Naming within LANs (with MAC addresses)
  - Sharing of medium
  - Flow control (in some cases)

- **LLC:**
  - Retransmission strategy
  - Link management
    - (deciding when a link exists or does not)
  - Flow control (in some cases)
Framing of Bits

- The objective is for the receiver to understand the packets (frames) sent by the sender when bits may get corrupted over the channel.
- Three approaches:
  - Character- or byte-oriented framing: BISYNC, IMP-IMP, SLIP, and PPP
  - Bit-oriented framing: HDLC
  - Clock-based framing: SONET
Character-Oriented Framing

- A frame starts and ends with a predefined sequence of control bytes or characters, and the occurrence of such sequence in the packet payload is avoided by byte stuffing.

- Consider the ARPANET IMP-IMP protocol:

<table>
<thead>
<tr>
<th>SYN</th>
<th>SYN</th>
<th>DLE</th>
<th>STX</th>
<th>packet payload (header and data)</th>
<th>DLE</th>
<th>ETX</th>
<th>CRC</th>
<th>CRC</th>
<th>CRC</th>
</tr>
</thead>
</table>

  SYN = synchronization
  DLE = data link escape
  STX = packet start

**Character stuffing:**
If DLE occurs in packet data, sender substitutes DLE with DLE DLE
Receiver substitutes DLE DLE in packet payload with DLE
In PPP, flag is 01111110 and such pattern in payload is preceded by 01111101

*Read Section 5.8 in textbook summarizing PPP*
Bit-Oriented Framing

- The same procedure is used with bits, bits are stuffed to break the occurrence of control flags.
- Bit stuffing consists of adding a bit after the occurrence of a bit pattern equal to the control flag used to frame the packet.
- Assume the control flag is 01111110
- If flag pattern occurs in payload, sender must transmit something different and receiver must be able to get original data.
- HOW?...Sender inserts a 0 after 5 1’s, and receiver deletes any 0 received after five 1’s in a bit sequence, and the frame is ended if after 5 1’s “10” is detected.

Original data: .....1011010100111111011000101111110011....
flag payload with bit stuffing flag

| 01111110 | .....101101010011111101100010111111010011.... | 01111110 |

Receiver then deletes stuffed 0 in any sequence ...111101 and obtains original data: .....1011010100111111011000101111110011....
Error Control

- *Section 5.2 of textbook*

- Majority of protocols use error detection coding because it requires fewer overhead bits than error correction coding.

- However, error detection is effective only if the bandwidth-delay product of the link is small, i.e., if feedback regarding errors in received packets can be given quickly to the sender, before it sends too many bits.

- Error detection code used by almost all networks is the *cyclic redundancy code*(CRC)
CRC codes detect errors by adding a few bits (redundancy bits or CRC bits) to each packet.

The attractive features of CRC are that only a few redundancy bits are needed to protect many bits of information, and it can be implemented with very simple hardware.

A message of \( m \) bits is transmitted with \( r \) redundancy bits as a transmitted string \( T = M.R \).

With the message bits being the most significant bits transmitted, and given that \( R \) occupies \( r \) bit positions, we have

\[
T = M \cdot 2^r + R
\]
The procedure to choose R to protect M is remarkably simple.

A string $G$ of $r + 1$ bits called the generator is agreed upon.

$R$ is chosen so that $T = A \times G$ for some $A$

$=>$ If the received string $T'$ is not a multiple of $G$, then an error has been detected!

$T'$ equals the (modulo 2) addition of $T$ and an error pattern $E$, such that a 0 in the pattern indicates no error in that bit position.

Now we have:

$$T \equiv M \cdot 2^r + R = A \times G; \text{ therefore,}$$

$$M \cdot 2^r = A \times G + R$$

Also $T' \equiv M \cdot 2^r + R + E$
CRC

- Therefore, to protect M, the sender:
  - Computes R by dividing M (shifted r bit positions) by G
  - Transmits M . R
- When the receiver obtains T’:
  - It divides T’ by G
  - Decides that there is an error if the remainder of the division is not 0
- The trick is then choosing a G for which the likelihood that T’ contains an error string E that is divisible by G is very small.
The choice of $G$ is done such that:
- First and last bit of $G$ are 1: This ensures that all single bit errors are detected.
- $G$ is a factor of $(x+1)$: To detect any odd number of errors.
- $G$ has a factor with at least three terms: To detect double-bit errors.

Code detects all burst of errors with fewer than $r$ bits and most other burst errors.

- e.g., Ethernet uses “CRC-32”
Types of MAC Protocols

- **Contention-based protocols**
- **Conflict-free:**
  - Fixed assignment (TDMA, FDMA)
  - Reservations
  - Polling
  - Token passing
Contestion-Based MAC Protocols

- **No coordination:** Stations transmit at will when they have data to send (e.g., ALOHA).
- **Carrier sensing (listen before transmit):** Stations sense the channel before transmitting a data packet (e.g., CSMA).
- **Listen before and during transmission:** Stations listen before transmitting and stop if noise is heard while transmitting (CSMA/CD).
- **Collision avoidance (floor acquisition):** Stations carry out a handshake to determine which one can send a data packet (e.g., MACA, FAMA, IEEE802.11, RIMA).
- **Collision resolution:** Stations determine which one should try again after a collision.
ALOHA Protocol

- The first protocol for multiple access channels; the first analysis of such protocols (Norm Abramson, Univ. of Hawaii, 1970).
- Originally planned for systems with a central base station or a satellite transponder.

Two frequency bands; Up link and down link (413MHz, 407MH at 9600bps)

*Central node retransmits every packet it receives!*
ALOHA Protocol

- Population is a large number of bursty stations.
- Each station transmits a packet whenever it receives it from its user; no coordination with other stations!
- Central node retransmits all packets (good or bad) on down link.
- Stations decide to retransmit based on the information they hear from central node.
An integral part of the ALOHA protocol is feedback from the receiver. Feedback occurs after a packet is sent. No coordination among sources.
The ALOHA Channel

We assume:

- An infinite population of stations.
- An ideal perfect down link for the transmission of feedback to senders.
- Stations are half duplex; have zero processing delays.
- Retransmissions are scheduled such that all packets are statistically independent.
- Each packet has the same duration $P$.
- Stations have the same round-trip delay from one other; this time can be much longer than $P$ (irrelevant).
- Collisions are the only sources of errors.
The ALOHA Channel

\[ \tau \text{ is not important} \]

What percentage of time is the channel sending correct packets?
Throughput of ALOHA Protocol

Node i’s frame is vulnerable from any arrival in the time interval (t0-1, t0+1)

All packets have the same length

packet overlaps with start of packet from node i

packet overlaps with end of packet from node i

interfering frame

node i frame

interfering frame
Throughput of ALOHA Protocol

Organize the time axis around packet from node i in terms of time slots lasting one packet length, and starting times given according to the start of the packet from node i.
The probability of a station starting a packet in a given time slot is $p$. Node $i$ transmits in time slot starting at $t_0$, i.e., time slot 2. The packet from node $i$ is successful if no other station transmits in the time slots 1 and 2.
Throughput of ALOHA Protocol

Because transmissions are independent, this means that the packet from Node $i$ succeeds with probability

$$P\{\text{pkt from node } i \text{ succeeds}\} = p(1-p)^M (1-p)^M = p(1-p)^{2M}$$

$M = N - 1$

Only one node can succeed, and there are $C(N, 1)$ ways to pick the winning node; hence, the probability that any given packet that starts at time $t_0$ succeeds is:

$$P\{\text{a pkt succeeds}\} = Np\left((1-p)^M\right)^2$$  \hspace{1cm} (a)
Throughput of ALOHA Protocol

\[ N_p \text{ is the total traffic generated in a given time slot; call it } G, \text{ i.e., } \]
\[ N_p = G \text{ and } p = G/N \]

The throughput of the channel is the percentage of packets offered to the channel that are successful.

The throughput of the channel then equals the probability that a packet is successful, i.e.,
\[ S = P\{a \text{ pkt succeeds}\} \]
Throughput of ALOHA Protocol

Substituting the above in Eq. (a) we obtain:

\[ S = G \left( 1 - \frac{G}{N} \right)^{N-1} \]

Make \( N \) tend to infinity (a large user population; \( N \sim N-1 \)).

A useful limit is:

\[ \lim_{x \to \infty} \left( 1 + \frac{1}{x} \right)^x = e^1 \]

Making \( x = N \), we have for a very large \( N \):

\[ G \left( 1 + \frac{(-G)}{N} \right)^N \to G(e^{-G})^2 \]

\[ S = Ge^{-2G} \]
Throughput of ALOHA Protocol

- Packet overlaps with start of packet from node i
- Packet overlaps with end of packet from node i
- Interfering frame

Node i’s frame is vulnerable from any arrival in the time interval (t₀⁻¹, t₀+1]

Highest throughput when we have one packet for each 2-packet time period
Slotted ALOHA

- The throughput of ALOHA can be improved by reducing the time a packet is *vulnerable* to interference from other packets.
- Slotted ALOHA works in a “slotted channel” providing discrete time slots.
- Stations can start transmitting only at the beginning of time slots.
- The time synchronization needed for slotting is accomplished at the physical layer, and some synchronization is required in many cases anyway.
Slotted ALOHA Protocol

Packet ready?  

- yes: Wait for start of next slot
  - transmit
  - wait for a round-trip time quantized in slots
    - yes: positive ack?
      - yes: compute random backoff integer k
      - no: delay packet transmission k times
    - no: delay packet transmission k times

- no: Wait for start of next slot
Slotted ALOHA

Propagation delay is part of time slot only in terrestrial nets
Throughput of Slotted ALOHA

The vulnerability period of a packet is a slot time:

Any arrivals in prior slot collide with packet $i$

The probability that a node transmits in a given time slot is $p$, and there are $N$ users. We then have:

$$P\{\text{pkt from any node succeeds}\} = \binom{N}{1}p(1-p)^{N-1} = Np(1-p)^{N-1}$$

With $G=NP$ and taking the limit as $N$ tends to infinity as we did before:

$$S = Ge^{-G}$$

We double the capacity of the channel (to about 36%) because we reduce in half the vulnerability period of a packet.
CSMA: Carrier Sense Multiple Access

- The capacity of ALOHA or slotted ALOHA is limited by the large vulnerability period of a packet.
- By listening before transmitting, stations try to reduce the vulnerability period to one propagation delay.
- This is the basis of CSMA (Kleinrock and Tobagi, UCLA, 1975)
- Same assumptions made for ALOHA are made now for CSMA.
Assume non-persistent carrier sensing.
Requires a maximum propagation delay much smaller than packet lengths!
CSMA Throughput

A virtual secondary channel used to send ACKs reliable and in 0 time!
Same assumptions made for pure ALOHA analysis.
All stations are at one propagation delay from each other and that equals: \( \tau \)

Peer-to-peer communication
No base station or transponder
Explicit feedback to sender!
CSMA Protocol

τ \text{ is critical!}

An interesting difference compared to ALOHA is that busy periods are bounded!

\begin{align*}
B & \leq P + 2\tau
\end{align*}
CSMA Throughput

- Because prop. delay is much smaller than packet length, slotted and pure CSMA have very similar performance.
- When MAC protocol requires small prop. delays, we can use slotted version to predict performance of unslotted version.

Reminder: These results are only an *upper bound* on performance, because we did not take into account the effect of ACKs sent from receivers!