CSE331: Introduction to Networks and Security — **Homework 1**

Fall 2006 — **Due: 29 Sept. 2006**

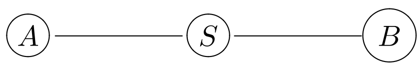
What to turn in: Submit hardcopy answers to the questions below. Please include your name, e-mail address, and the number of hours you spent working on the assignment.

1. (20 points)

Consider a point-to-point link 2km in length. At what bandwidth would propagation delay (at a speed of 2×10^8 m/sec) equal transmit delay for 100-byte packets? What about 512-byte packets?

- 100-byte packets:
  transmit =
  $$\frac{800 \text{ b}}{X \text{ b/sec}}$$
  propagation =
  $$\frac{2000 \text{ m}}{2 \times 10^8 \text{ m/sec}}$$
  Setting the two ratios above to be equal, we have:
  $$\begin{aligned} 2000 \times X &= 1600 \times 10^8 \\ X &= 0.8 \times 10^8 \end{aligned}$$
  Or, 80Mb/sec.
- 512-byte packets:
  transmit =
  $$\frac{4096 \text{ b}}{X \text{ b/sec}}$$
  propagation =
  $$\frac{2000 \text{ m}}{2 \times 10^8 \text{ m/sec}}$$
  Setting the two ratios above to be equal, we have:
  $$\begin{aligned} 2000 \times X &= 8192 \times 10^8 \\ X &= 4.096 \times 10^8 \end{aligned}$$
  Or, 409.6MB/sec

2. (20 points)



Hosts A and B are each connected to a switch S via 10-Mbps links as shown above. The propagation delay on each link is 20μs. S is a store-and-forward device that can send and receive bits simultaneously; it begins retransmitting a received packet 35μs after it has finished receiving it (if it can). Calculate the total time in milliseconds required to transmit 12,000 bits from host A to host B

a. As a single packet.

Each link contributes a time of

$$20\mu\text{s} + 12 \times 10^3 \text{ bits}/(10 \times 10^6 \text{ bits/sec}) = 20\mu\text{s} + 12 \times 10^{-4} \text{ sec} = 1.22 \text{ ms}$$

The total times is thus

$$2 \times 1.22 \text{ ms} + 0.035 \text{ ms} = 2.475 \text{ ms}$$

b. As three 4,000 bit packets sent one right after the other.

The two packets take less time. Note that a 4000 bit packet takes 4000 bits/(10^7 bits/sec) $=$ 0.4ms to transmit (not including the propagation delay).

Host A begins sending the first packet at time 0, and it begins arriving at host S at time 0.02ms. S finishes receiving the first packet at time 0.42msand begins retransmitting the first packet after 35μs, which is time 0.455ms. Host B begins receiving the first packet after 20μs (at time 0.475). Simultaneously, the second packet has been arriving at S. At time 0.82ms, host S finishes receiving the rest of the second packet, but can't yet retransmit it because it is still sending the first packet. S finishes sending the first packet at time 0.855 and starts transmitting the second packet. The third packet follows similarly, so the total time taken is 1.655 ms

3. (20 points)

The utility program `ping` can be used to estimate the RTT to various Internet hosts. It is available on both Unix systems (`/bin/ping`) and Windows (`c:/WINNT/system32/ping`).

The Unix program `traceroute` (or the Windows equivalent `tracert`) can be used to find the sequence of routers through which a message is routed.

Using `ping` and `traceroute`, estimate the RTT and number of hops to the following hosts. How well does RTT correlate with number of network hops? How well does the number of hops correlate with geographical distance? Test at least three additional hosts to support your claims. Note that your results may vary depending on the time of day and location of the host you use to run the experiment. (Why?)

Answers were determined by running the experiment on `blue.seas.upenn.edu`. RTT does not necessarily correlate well with the number of hops—consider a single hop over a long link versus a number of hops over a fast Ethernet link. The number of hops does not necessarily correlate well with the geographic proximity—the number of hops from a node on Penn's campus to nodes in Philadelphia, California, and Japan range from 18–21.

Time of day influences the answers because Internet traffic (i.e. congestion) may mean more delays and retransmissions—there is more traffic during business hours. (People are awake and working or surfing!)

To get full credit, you must have checked other likely candidates to test your hypothesis.

a. `eniac.cis.upenn.edu`

RTT is approx. 0 ms. # hops is approx. 1.

b. `www.upenn.edu`

RTT is approx. 3 ms. # hops is approx. 4.

c. `www.merck.com` (in New Jersey)
RTT is approx. 95 ms. # hops is approx. 16.

d. `www.cam.ac.uk` (in England)
RTT is approx. 100 ms. # hops is approx. 19.

e. `www.kyoto-u.ac.jp` (in Japan)
RTT is approx. 230 ms. # hops is approx. 21.

4. (20 points)

Suppose we want to transmit the message 11001001 and protect it from errors using the CRC polynomial $z^3 + 1$.

a. Use polynomial long division to determine the message that should be transmitted. Show your work.

```
        11010011

       ------------
1001  |11001001000
        1001
        ----
         1011
         1001
         ----
           1000
           1001
           ----
              1100
              1001
              ----
               1010
               1001
                011  <--- Remainder
```

Transmitted Message = 11001001011

b. Suppose the first bit of the message is inverted due to noise on the transmission link. What is the result of the receiver's CRC calculation? Show your work.

```
        01000001
       ------------
1001  |01001001011
        0000
        ----
         1001
         1001
         ----------
             001011
               1001
               ----
                010  <-- Remainder
```

5. (20 points)

Let A and B be two stations attempting to transmit on an Ethernet. Each has a steady queue of frames ready to send. A's frames are numbered A_1, A_2, and so on; B's frames are numbered similarly. Let $T = 51.2\mu$s be the exponential backoff base unit.

Suppose A and B simultaneously attempt to send frame 1, collide, and happen to choose backoff times of $0 \times T$ and $1 \times T$, respectively, meaning A wins the race and transmits A_1 while B waits. At the end of this transmission, B will attempt to retransmit B_1 while A will attempt to transmit A_2. These attempts will collide, but now A backs off for either $0 \times T$ or $1 \times T$, while B backs off for time equal to one of $0 \times T, \ldots, 3 \times T$.

a. Give the probability that A wins this second backoff race immediately.

For A to win immediately, it must choose to delay for $0 \times T$ seconds, which it does with probability $\frac{1}{2}$. For B not to collide with this event, it must choose to wait for $1 \times T$ or longer, which it does with probability $\frac{3}{4}$. Multiplying, we get a probability of $\frac{3}{8}$ that A wins immediately.

(Note that A can still win when it chooses $1 \times T$ if B chooses a time $> 1 \times T$. This will occur with probability $\frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$. So, A will win with probability $\frac{3}{8} + \frac{1}{4} = \frac{5}{8}$)

b. Suppose A wins the second backoff race. A transmits A_3, and when it is finished, A and B collide again as A tries to transmit A_4 and B tries once more to transmit B_1. Give the probability that A wins this third backoff race immediately.

As above, A must choose delay of $0 \times T$, which it does with probability $\frac{1}{2}$. B must choose a delay $> 0 \times T$, which it now does with probability $\frac{7}{8}$, so the net probability that A wins immediately this time is $\frac{7}{16}$.

(Note that, as above, A still wins with probability $\frac{3}{8}$ even if it chooses $1 \times T$, so the total probability that it wins is $\frac{13}{16}$.)

c. Give a reasonable lower bound for the probability that A wins the remaining backoff races.

In general, for the n^{th} race, the probability that A wins round n is given by the formula:

$$P[A\ wins\ n] = \frac{1}{2}(P[C_n(B) > 0]) + \frac{1}{2}(P[C_n(B) > 1])$$

Where $P[C_n(B) > x]$ is the proability that B's n^{th} backoff choics is greater than x. Since, for the n^{th} race, B chooses a number > 0 with probability $P[C_n(B) > 0] = \frac{2^n-1}{2^n}$ and a number > 1 with probability $P[C_n(B) > 1] = \frac{2^n-2}{2^n}$. We have that A will win the n^{th} race with probability:

$$P[A\ wins\ n] = \frac{1}{2}(\frac{2^n-1}{2^n}) + \frac{1}{2}(\frac{2^n-2}{2^n}) = \frac{2^n-1}{2^{n+1}} + \frac{2^n-2}{2^{n+1}} = \frac{2^{n+1}-3}{2^{n+1}}$$

(We can see that the first two answers follow this general case for $n = 2$ and $n = 3$.)

To get an exact lower bound on A winning the remaining rounds, we would need to calculate:

$$P[A\ wins\ 4] \times P[A\ wins\ 5] \times \ldots \times P[A\ wins\ 9] \times (P[A\ wins\ 10]^7)$$

The reason why the last term is $(P[A\ wins\ 10]^7$ is that Ethernet implementations typically try for 16 total rounds, but put a limit of $2^{10} \times T$ on the time their willing to wait.

One could calculate this number exactly, but it suffices to observe that each term in this sequence is getting slightly larger, so to calculate a reasonable lower bound, we could just take $P[A\ wins\ 4]^{13}$, which yields a lower bound probability of approximately 0.28. A better estimate is $P[A\ wins\ 4] \times P[A\ wins\ 5]^{12} \approx 0.51$. An even better estimate is $P[A\ wins\ 4] \times P[A\ wins\ 5] \times P[A\ wins\ 6]^{11} \approx 0.67$, etc.

The exact calculation is:

$$\frac{29}{32} \times \frac{61}{64} \times \frac{125}{128} \times \frac{253}{256} \times \frac{509}{512} \times \frac{1021}{1024} \times (\frac{2045}{2048})^7 \approx 0.82$$

d. What happens to frame B_1? (This is known as the Ethernet *capture effect.*)

B_1 never gets delivered with probability 0.82. A has "captured" the link, and B is unlikely to win any of the backoff races.