# HW #3

ELEN E6761 - Computer Communication Networks — Due 11/14/2000, 5pm (on-campus and CVN)
Fall 2000 — Prof. Rubenstein

Please turn your homework into Vassilis Stachtos' mailbox, E2 (oustide of the Electrical Engineering office, fax it by 5pm, or make some other arrangement with Vassilis. On-campus students can fax to Professor Rubenstein at (212) 932-9421 (remember to include your name and that the fax is for Professor Rubenstein). CVN students can fax as above, or fax to the CVN fax number. LATE ASSIGNMENTS WILL HAVE THEIR GRADE REDUCED BY 10% EACH BUSINESS DAY IN WHICH THEY ARE LATE!!

1. Assume a set of $n$ receivers, labelled $r_1, \cdots, r_n$, reside on an unbridged LAN. Each receiver transmits a fraction of time $p$ (e.g., if $p = .2$, then the receiver transmits 20% of the time.) A transmission of a frame takes time $T$. Let $X_A(i)$ be a random variable that equals 1 if, using algorithm $A$, $r_1$'s $i$th transmission is successful (i.e., does not collide with any other receiver transmissions.) Compute $P(X_A(i) = 1)$ for each of the following algorithms (Note: solutions should have no dependence on $i$)

   (a) $A = 1$: $A$ is Slotted Aloha

   (b) $A = 2$: $A$ is Aloha, and each receiver transmits periodically with every $T/p$ seconds, where the start time of the transmission is chosen uniformly by each receiver. Note: this is a dumb protocol, since two receivers that collide will collide every time, but it makes the problem much easier to solve. Hint: Draw a circle...

   (c) $A = 3$: $A$ is Aloha, where each receiver waits a random amount of time that is exponentially distributed with rate $\frac{p}{T(1-p)}$. Hint: Apply PASTA when $r_1$ decides to transmit, then use the memoryless nature of the exponential distribution used by the other receivers. (Note: when $p = 1/n$, then $P(X_3 = 1) = P(X_1 = 1)/e$.)

2. Assume that the sender's and receiver's clocks are skewed (running at different rates and unsynchronized). Let $R(t) = At + B$ be the time returned by the receiver's clock when the sender's clock has time $t$, where $A$ and $B$ are constants. For jitter calculations to be accurate, what restrictions (if any) must there be on the values of $A$ and $B$?

3. Prove by induction that a single-bit parity check can always detect an odd number of bit-flips

$$\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 1 & 0 & 0 & 0 \end{array}$$

4. Consider the 2-D parity check above. Assume bits are incorrect with probability $p < .5$. What is the most likely set of bits that are flipped (give row, column coordinates)?

5. What is the minimum number of bits that need to be flipped in a 2-D parity check such that a corrupt packet can be accepted? Prove your answer.

6. Consider a packet-level forward error correction scheme that uses only XOR operations between sets of packets to generate repair packets. Show (by induction on the number of packets used in the encoding) that if $n$ arbitrary packets are used in the encoding, then it is impossible to retrieve all original data packets if less than $n$ packets are receieved.