Solution 4 (EECS 333) Introduction to Communication Networks

Prof. Dongning Guo

April 24, 2007

Problem 1  FEC vs. ARQ

1. Note the pattern for a \((n,k)\) Hamming code is \(n = 2^m - 1\), \(k = 2^m - m - 1\), where \(m\) is the number of parity bits. And one can easily check \(d_{\text{min}} = 3\) which is true for all Hamming codes. This guarantees the capability of correcting single bit error. In order to make \(k \geq 1000\), one needs \(m \geq 10\). That is, a \((1023,1013)\) Hamming code should be used in which redundant bits are filled to make the number of data bits 1013. Therefore, the redundancy is \(23/1023 = 0.0225\).

2. 1 (single parity check bit suffices).

3. \(1 + (\frac{1}{1-p} - 1) \times 1001\).

4. \(1 + (\frac{1}{1-p} - 1) \times 1001 \leq 23 \Rightarrow p \leq 0.0215\).

Problem 2  Parallel ARQ

Solutions follow questions:

a. Qualitatively, compare the relative performance of this protocol with Go-Back-N ARQ and with Stop-and-Wait ARQ.

For simplicity assume that the time between consecutive frame transmissions in Stop-and-Wait corresponds to \(N\) consecutive transmissions without stopping. The parallel Stop-and-Wait procedure described above is an effective way to fill the transmission pipe without the additional complexity of Go-Back-N ARQ.

Vs. Go-Back N. Go-Back-N delivers frames in order. The parallel Stop-and-Wait protocol does not deliver frames in order, so additional processing is required if frames must be delivered in sequence. Because all the processes are independent, this protocol retransmits erroneous frames individually. In contrast, the Go-Back-N protocol retransmits a group of \(N\) frames. In this sense, the parallel protocol seems to perform similarly to a Selective Repeat process.

Vs. Stop-And-Wait. If Stop-and-Wait is used, the effective bit rate, without errors, will be \(N\) times less than the protocol described here. In fact, the larger \(N\), the more efficient the protocol described is. At its worst case, where \(N = 1\), it reduces to Stop-And-Wait.

b. How does the service offered by this protocol differ from that of Go-Back-N ARQ?

The parallel Stop-and-Wait protocol does not deliver frames in order, unless augmented by a frame resequencing scheme.
Problem 3  Finite State Machine

Suppose $t$ is the transmission time for one unit.

1. Let $T_1$ denote the time for going from state $(0,0)$ to $(0,1)$. Start from $(0,0)$, the state transits to $(0,1)$ when frame 0 is correctly received. On average, the number of retransmissions needed is given by $\left(\frac{1}{1-P_f}-1\right) = \frac{P_f}{1-P_f}$. Therefore, the average time $E[T_1] = 2t + 3t\frac{P_f}{1-P_f}$.

2. Let $T_2$ denote the time for going from state $(0,1)$ to $(1,1)$. This happens when ACK 0 is correctly received. Note if an ACK error occurs, the receiver needs correctly receive another frame 0 (which will be retransmitted by the transmitter due to timeout) before it initiates a new ACK 0. Similarly, the average number of retransmissions is $P_a$. The time between two retransmissions is $E[T_1] + t$. Therefore, $E[T_2] = \frac{P_a}{1-P_a}(E[T_1] + t) + t$.

3. Throughput equals $\frac{1}{E[T_1] + E[T_2]}$ (frames/sec).

Problem 4  Selective Repeat ARQ

Assume ACK frame arrives at the transmitter slightly before the next data frame transmission.

Problem 5  ALOHA

Maximum throughput for ALOHA = 0.184.

Maximum throughput in frames/sec = \(\frac{56000 \text{ bits/sec}}{1000 \text{ bits/frames}} \times 0.184 = 10.304 \approx 10\).

Problem 6  3-node ALOHA

In slotted ALOHA system, a slot is successful if and only if there is one node transmitting.

1. The probability $P_A$ of node A being successful in one slot is: $P_A = (1-p)^2 p$. A succeeds for the first time in slot 4 means A fails or does not attempt to transmit during the first 3 slots and succeeds in slot 4. Therefore, the probability $P_1 = (1 - P_A)^3 P_A$.

2. $P_2 = 3P_A$.
3. $P_3 = (1 - P_2)^3 P_2$.
4. Efficiency = $P_2 = 3p(1 - p)^2$. The plot is on the next page.