Homework 2 Solutions

85 points total

Exercises from Chapter 2:

5 (5 points)
The stuffed bits (zeros) are underlined:
1101 0111 1100 1011 1110 1010 1111 1011 0

6 (5 points)
The ∧ marks each position where a stuffed 0 bit was removed. There were no stuffing errors detectable by
the receiver; the only such error the receiver could identify would be seven 1’s in a row.
1101 0111 1110 1111 1A010 1111 1A110

18 (10 points)
(a) We take the message 11100011, append 000 to it, and divide by 1001 according to the method shown
in Section 2.4.3. The remainder is 100; what we transmit is the original message with this remainder
appended, i.e., 1110 0011 100.
(b) Inverting the first bit of the transmission gives 0110 0011 100; dividing by 1001 gives a remainder of
10; the fact that the remainder is nonzero tells us an error has occurred.

23 (15 points)
(a) Propagation delay = 40 × 10³m/(2 × 10⁸m/s) = 200 μs.
(b) The roundtrip time would be 400μs (0.4ms). A plausible timeout value would be twice this, or 0.8ms.
Smaller values (but larger than 0.4ms!) might be reasonable, depending on the amount of variation in
actual RTTs.
(c) The propagation-delay calculation does not consider processing delays that may be introduced by the
remote node; it may not be able to answer immediately.

24 (5 points)
Bandwidth × RTT is 1Mbps * 2 * 1.25s = 2.5Mb = 312 KB, or 312 frames. The send window size should
be this large; the sequence number space must cover twice this range, or up to 624. So 10 bits are needed.

35 (15 points)
(a) The smallest working value for MaxSeqNum is 8. It suffices to show that if DATA[8] is in the receive
window, then DATA[0] can no longer arrive at the receiver. Suppose we have DATA[8] in receive
window, then the following are implied:
• The earliest possible receive window is DATA[6] .. DATA[8] because RWS=3
• DATA[0] .. DATA[5] have been received by the receiver.
• ACK[0] has been received by the sender, so the sender will not retransmit DATA[0], i.e.,
DATA[0] can no longer arrive at the receiver.
(b) We show that if MaxSeqNum=7, then the receiver can be expecting DATA[7] and DATA[0] can still
arrive.
1. Sender sends DATA[0] .. DATA[4]. All arrive.
2. Receiver sends ACK[0] .. ACK[4] in response, but all are lost. The receive window is now
DATA[5] .. DATA[7].
3. Sender times out and retransmits DATA[0] .. DATA[4]. The receiver accepts DATA[0] as
DATA[7], because they have the same sequence number.
(c) MaxSeqNum ≥ SWS + RWS.

43 (10 points)
(a) A can choose $k_A = 0$ or $1$; B can choose $k_B = 0, 1, 2, 3$. A wins if $(k_A, k_B)$ is among $(0, 1), (0, 2), (0, 3), (1, 2), (1, 3)$; there is a $5/8$ chance of this.
(b) Now A can choose $k_A = 0$ or $1$, and B can choose $k_B$ among $0..7$. If $k_A = 0$, there are 7 choices for $k_B$ that have A win; if $k_A = 1$ then there are 6 choices for $k_B$ that have A win. So the probability of A’s winning is $13/16$.

54 (10 points)
Whereas in wired networks, the sender can detect collisions as they occur, this is not true in wireless networks. There are two reasons for this. First, wireless nodes cannot transmit and receive at the same time. Second, a wireless node cannot receive transmissions from another wireless node if they are not within range of each other.

Additional problems:

1. (5 points)
The minimum frame size is equal to the number of bits a node can transmit in RTT (i.e., the worst case collision detection time). So the minimum frame size = bandwidth * RTT = 100Mbps * 51.2 µs = 5120 bits = 640 bytes.

2. (5 points)
802.11 uses the RTS/CTS mechanism to address the hidden node problem. Consider the scenario shown in Figure 2.30. Suppose both A and C want to send data to B; both would send RTS to B. B only responses to one RTS, say A’s RTS. C sees the CTS sent from B to A, so it won’t send to B, thus avoiding collision at B.