1.

a. Worst Case: k-N ~ k, sender not get ACK from k-N to k

Best Case: k~ k-1+N, sender get ACK from k-N to k-1

Thus, possible sets of sequence number are

[k-N, k] ~ [k-1, k-1+N] for (k – N >= 0 and k + N <= 4096)

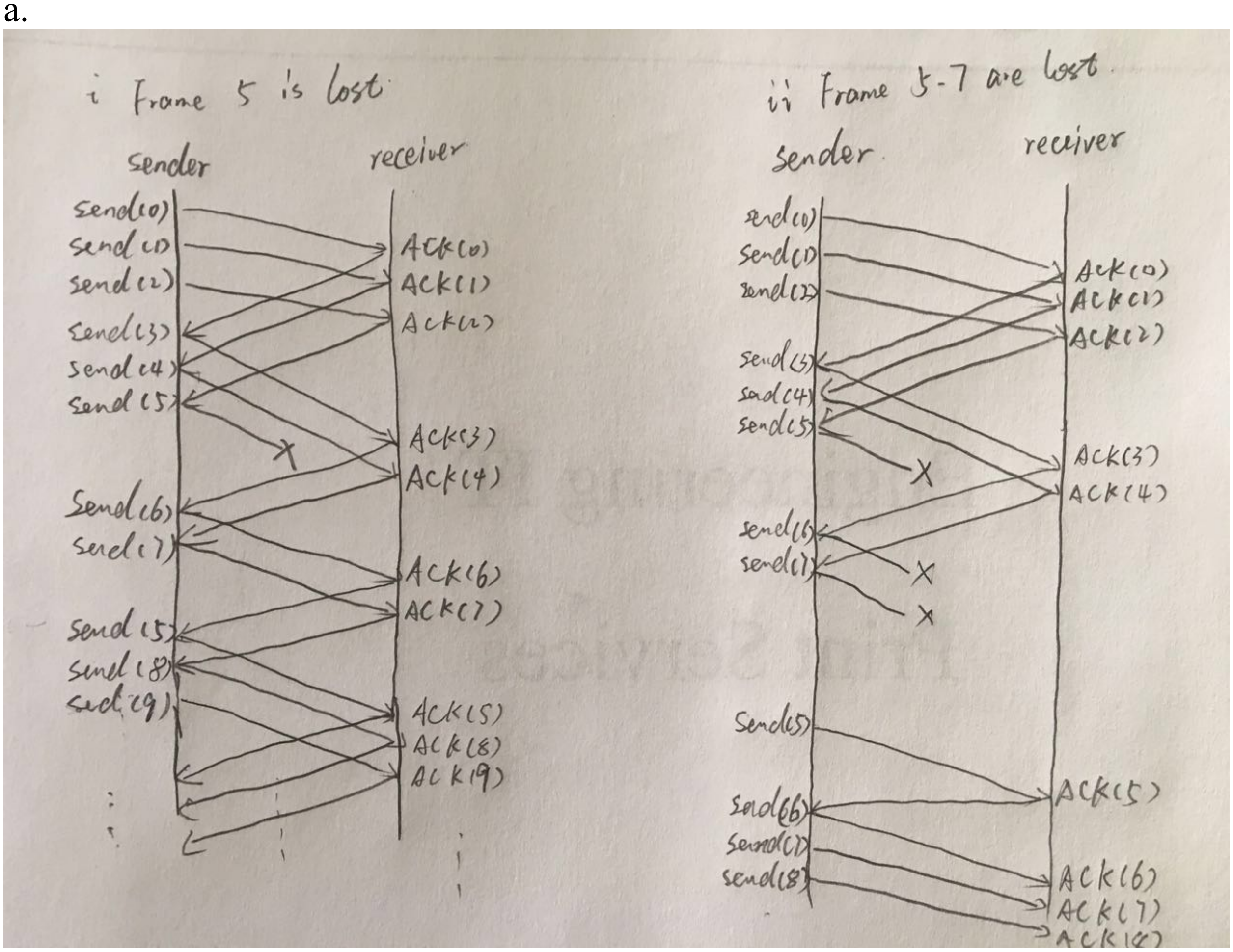
b. [k-N, k-1] for (k-N>=0 and K-1 <=4096)

Since receiver is expecting sequence number k, all values of ACK from last window size can be possible propagating.

c. It is possible. piazza@873 When the receiver receives a packet that is outside its receive window, receiver drop the packet and send an ACK for the last received frame in the window. Thus, it is possible the sender gets an ACK for last received frame which packet (k-1) falls outside current window.

2.

a.



b. Assume sender sends packet 4,5,6,7,8, receiver receives these packets successfully. But ACKs of them are lost. So for now, receiver is looking forward to get packet 0,1,2,3,4, but sender would send 4,5,6,7,8 again since no ACK and time-out. Now, receiver is looking forward to get a new packet 4 but it gets old packet 4.

3.

a. 120000/(1500-80) = 85 packets
   (1500*8)/100000000 = 0.00012 s = 0.12 ms
   (150*8)/100000000 = 0.000012 s = 0.012 ms
   (0.12 ms + 6ms + 0.12ms + 0.012ms + 6ms + 0.012ms) * 85 = <u>1042.44 ms</u>

b. Throughput = TransferSize / TransfertTime
   = 120000 bytes * 8 / 1042.44 ms = 0.920916312 Mbps

c. 120000/(1500-80) = 85 packets
   (1500*8)/100000000 = 0.00012 s = 0.12 ms
   (150*8)/100000000 = 0.000012 s = 0.012 ms
   85/20 = 4.25 → 5
   (0.12 ms + 6ms + 0.12ms + 0.012ms + 6ms + 0.012ms) * 5 = <u>61.32 ms</u>
   120000 bytes * 8 / 61.32 ms = <u>15.66 Mbps</u>

d. bandwidth-delay product = 2 * 100 Mbps * 0.012 s = 2.4 MB
   SWS = RWS = 2 * 2.4 = 4.8 → 5
   85/5 = 17
   (0.12 ms + 6ms + 0.12ms + 0.012ms + 6ms + 0.012ms) * 17 = 208.488 ms
   120000 bytes * 8 / 208.488 ms = <u>4.6 Mbps</u>

e. SWS = RWS = 3 * 2.4 = 7.2 → 7
   85/7 = 12.14 → 13
   (0.12 ms + 6ms + 0.12ms + 0.012ms + 6ms + 0.012ms) * 13 = 159.432 ms
   120000 bytes * 8 / 159.432 ms = <u>6.02 Mbps</u>

4.

a. min(1.5Gbps, 60Mbps, 85Mbps) = <u>60Mbps</u>

b. speed in fiber: 2 * 10^8 m/s
   2*(300km + 35km + 25km) / 2 * 10^5 km/s = <u>0.0036 s</u>

c. 60 Mbps * 0.0018 s = <u>$1.08 * 10^5$ bits</u> -2

d. $(1.08 * 10^5 * 2) / (1024 * 8) \approx \underline{26}$
Thus, optimal value of SWS is 26.

e. Network becomes busy and receiver may drop more packets if we use an SWS value many times larger than the value you suggested in part (d).

5.
a. SRTT(15) = 0.7* SRTT(14)+(1-0.7) * RTT(15)
= 0.7 * SRTT(14) + 0.3 * 0.5s --- all RTT = 0.5s
= 0.7 * (0.7 * SRTT(13) + 0.15s) + 0.15s
….
$= 0.7^{15} * SRTT(0) + 0.15s + 0.15s * 0.7 + 0.15s * 0.7^2 \ldots + 0.15 * 0.7^{14}$
$= \underline{0.5024\ s}$

b.
$\alpha = 0.5$
$\rightarrow 0.5^{15} * SRTT(0) + 0.5*0.5s + 0.5*0.5s*0.5 + 0.5*0.5s*0.5^2 \ldots 0.5*0.5s*0.5^{14}$
$= \underline{0.500\ s}$

$\alpha = 0.9$
$\rightarrow 0.9^{15} * SRTT(0) + 0.1*0.5s + 0.1*0.5s*0.9 + 0.1*0.5s*0.9^2 \ldots 0.1*0.5s*0.9^{14}$
$= \underline{0.6029\ s}$

c.
Retransmission ambiguity problem: The ACK can be a response to first transmission or re-transmission. It's ambiguity that the ACK response which transmission.

Karn-Partridge algorithm avoid this ambiguity by ignoring retransmitted segments when updating the round-trip time estimate.

6.
a. $(2 * 10^9 * 0.05) / (1024*8) \approx$ 12200 segment

b. $(2 * 10^9 * 0.05) / (20*1024*8) \approx$ 610 segment

-4