

# Internet Infrastructure and Protocols (COMP5311)

Solutions to Assignment Three (due on 29 Nov 2010)

Each question carries 8 marks, unless stated otherwise.

Rocky K. C. Chang

- 1) (TCP congestion control) Referring to Figure 1 for TCP packet transmissions in a TCP Reno connection, a number of full-sized TCP data segments were sent from a sender to a receiver, and the segments are numbered starting from 0. Each ACK acknowledges only one TCP data segment. The square symbol refers to a TCP data segment transmission, whereas a small dot refers to an ACK transmission. The TCP data segment and its ACK are drawn on the same line for easy reference (an example is shown for segment 10). TCP Reno implements the fast retransmission and fast recovery algorithms that we discussed in class. The figure shows that only segment 14 was lost, and it was fast retransmitted after receiving a third duplicate ACK. Note that there were a number of duplicate ACKs received. The send window was always given by the  $cwnd$ , and the sender always had data to send.

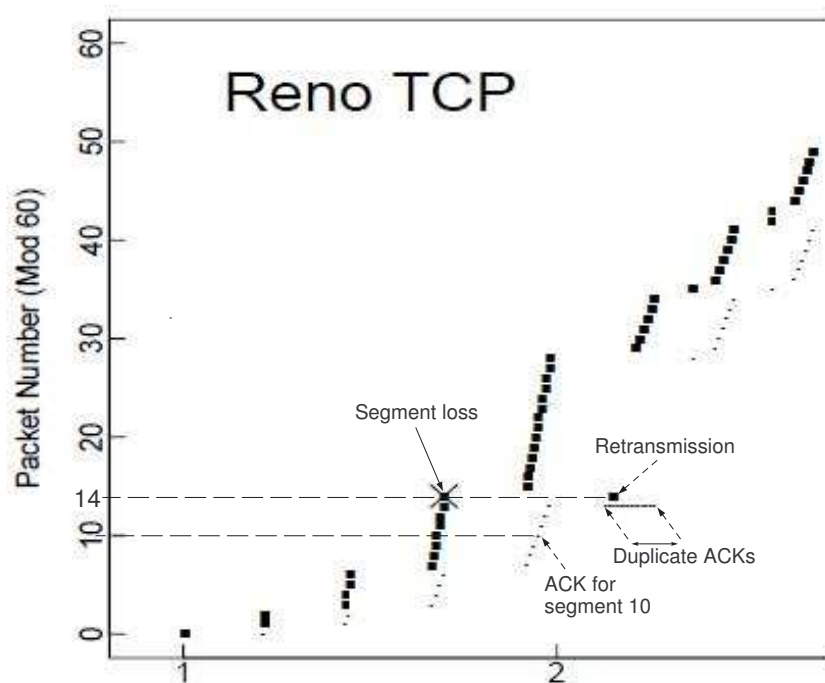


Fig. 1. Packet transmission sequence of a TCP Reno connection.

Answer the following questions about the trace in Figure 1.

- (2 marks) How many ACKs were received for segment 13? Explain your answer.
- (2 marks) What was the size of the  $cwnd$  (in terms of the number of data segments) immediately after starting the fast recovery phase? Explain your answer.
- (2 marks) At what time (in terms of an event) was the fast recovery phase ended? Explain your answer.
- (2 marks) At the time that the fast recovery phase was ended, what was the size of the  $cwnd$ ? Explain your answer.

**Solution:**

- (2 marks) There were 1 ACK and 14 duplicate ACKs received for segment 13. The duplicate ACKs were induced by out-of-ordered data segments 15-28.
- (2 marks) When the fast recovery phase was started, the sender set its  $cwnd$  to  $\min\{\text{flightsize}/2, 2MSS\}$ . The flightsize is 15; therefore,  $cwnd = 15/2 = 7.5$ .
- (2 marks) The fast recovery phase was ended when receiving the ACK for segment 28 which occurred around 2.35 seconds.

- d) (2 marks) The `cwnd` is reset to `sssthresh` which is 7.5 (round off to 7). This is why the sender could send only one more segment, because the ACKs for the other 6 had not been received yet.