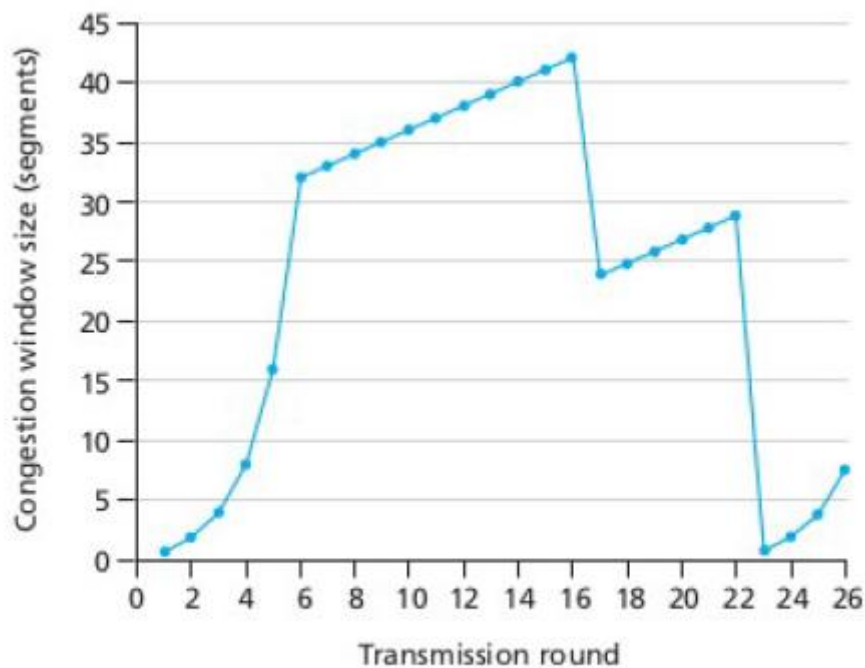


P40. Consider Figure 3.58. Assuming TCP Reno is the protocol experiencing the behavior shown above, answer the following questions. In all cases, you should provide a short discussion justifying your answer.

- Identify the intervals of time when TCP slow start is operating.
- Identify the intervals of time when TCP congestion avoidance is operating.
- After the 16th transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?
- After the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?
- What is the initial value of ssthresh at the first transmission round?
- What is the value of ssthresh at the 18th transmission round?
- What is the value of ssthresh at the 24th transmission round?
- During what transmission round is the 70th segment sent?
- Assuming a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, what will be the values of the congestion window size and of ssthresh?
- Suppose TCP Tahoe is used (instead of TCP Reno), and assume that triple duplicate ACKs are received at the 16th round. What are the ssthresh and the congestion window size at the 19th round?
- Again suppose TCP Tahoe is used, and there is a timeout event at 22nd round. How many packets have been sent out from 17th round till 22nd round, inclusive?



3.58 • TCP window size as a function of time

- (a) بازه های [1, 6] و نیز [23, 26] در TCP slow start قرار دارد چراکه اندازه پنجره به صورت نمایی در حال افزایش است.
- (b) بازه های [6, 16] و نیز [17, 22] در TCP congestion avoidance قرار دارد چراکه اندازه پنجره به صورت خطی در حال افزایش است.
- (c) با سه مرتبه ack تکراری مواجه شده ایم چرا که در صورت مواجه شدن با timeout می بایست اندازه پنجره ۱ تنظیم شده باشد اما در راند ارسال ۱۶ ام اندازه پنجره ۱ نشده بلکه تقریباً نصف شده است.
- (d) با timeout مواجه شده ایم چراکه اندازه پنجره برابر ۱ تنظیم شده است.
- (e) در مرحله slow start اندازه پنجره تا رسیدن به ssthresh به صورت نمایی افزایش می یابد و لذا با توجه به اینکه اندازه پنجره تا حدود ۳۲ افزایش داشته لذا در ابتدای امر ssthresh برابر ۳۲ بوده است.
- (f) زمانی که packet loss رخ می دهد و سه مرتبه ack تکراری دریافت می شود مقدار ssthresh برابر نصف اندازه پنجره تنظیم می شود در راند ۱۶ ام که packet loss رخ می دهد اندازه پنجره ۴۲ است لذا ssthresh در راند ۱۸ ام برابر ۲۱ خواهد بود.
- (g) در راند ۲۲ ام packet loss رخ داده و با timeout مواجه شده ایم و اندازه پنجره در این راند برابر ۲۹ بوده است لذا در راند ۲۴ ام می بایست مقدار ssthresh برابر نصف این مقدار باشد یعنی برابر ۱۴.
- (h) در هر راند به اندازه پنجره بسته ارسال می شود لذا در راند اول یک بسته در راند دوم ۲ بسته در راند سوم ۴ بسته و ... ارسال می شود لذا به این ترتیب می توان گفت بسته ۷۰ ام در راند ۷ ام ارسال می شود.
- (i) با توجه به اینکه در راند ۲۶ در مرحله slow start هستیم چنانچه سه مرتبه ack تکراری دریافت کنیم به مرحله fast recovery رفته و ssthresh برابر نصف اندازه پنجره در راند ۲۶ ام یعنی برابر ۴ خواهد شد. و اندازه جدید پنجره برابر ۳ + ssthresh یعنی برابر ۷ خواهد شد.
- (j) مقدار ssthresh برابر ۲۱ تنظیم می شود چراکه در هر دو مورد TCP Reno و TCP Tahoe می بایست ضمن دریافت ۳ مرتبه ack تکراری مقدار ssthresh برابر نصف اندازه پنجره تنظیم گردد اما اندازه پنجره در راند ۱۷ ام برابر ۱ تنظیم می شود. و در راند ۱۸ ام برابر ۲ و در راند ۱۹ ام برابر ۴ خواهد شد.
- (k) در راند ۱۶ ام timeout رخ داده و لذا در راند ۱۷ ام ssthresh برابر $21 = 42/2$ تنظیم می شود و به همین ترتیب در راند ۱۷ ام یک بسته در راند ۱۸ ام دو بسته در راند ۱۹ ام ۴ بسته در راند ۲۰ ام ۸ بسته در راند ۲۱ ام ۱۶ بسته و در راند ۲۲ ام ۲۱ بسته ارسال می شود (به دلیل رسیدن به مقدار ssthresh) لذا تعداد کل بسته های ارسال شده در راند ۱۹ تا ۲۲ ام برابر $21 + 16 + 8 + 4 + 2 + 1 = 52$ می باشد.

3.P46. Consider that only a single TCP (Reno) connection uses one 10Mbps link which does not buffer any data. Suppose that this link is the only congested link between the sending and receiving hosts. Assume that the TCP sender has a huge file to send to the receiver, and the receiver's receive buffer is much larger than the congestion window. We also make the following assumptions: each TCP segment size is 1,500 bytes; the two-way propagation delay of this connection is 150 msec; and this TCP connection is always in congestion avoidance phase, that is, ignore slow start.

- What is the maximum window size (in segments) that this TCP connection can achieve?
- What is the average window size (in segments) and average throughput (in bps) of this TCP connection?
- How long would it take for this TCP connection to reach its maximum window again after recovering from a packet loss?

(a)

$$MSS = 1500 \text{ byte}$$

$$RTT = 150 \text{ msec}$$

$$(W * MSS) / RTT = 10 \text{ Mbps} \rightarrow$$

$$W * 1500 * 8 / 0.15 = 10 * 10^6 \rightarrow W = (10^7 * 0.15) / (8 * 1500) = 125$$

(b) اندازه پنجره از W تا $W/2$ تغییر می کند ولذا W_{avg} برابر $(W + 0.5W) / 2 = 0.75W$ می باشد و average throughput نیز برابر $W_{avg} * MSS / RTT$ می باشد یعنی می توان نوشت :

$$W_{avg} = 0.75W = 0.75 * 125 = 94$$

$$\text{Throughput}_{avg} = W_{avg} * MSS / RTT = 94 * 1500 * 8 / 0.15 = 7.52 \text{ Mbps}$$

(c) با توجه به صورت سوال چون از مرحله slow start صرف نظر شده است لذا در صورت رخداد packet loss اندازه پنجره نصف و حدود ۶۲ می شود و با توجه به اینکه می بایست مجدداً از این اندازه نصف شده به طور خطی به مقدار اولیه یعنی ۱۲۵ بازگردیم لذا می بایست ۶۳ راند طی شود و در هر راند اندازه پنجره به اندازه یک MSS افزایش پیدا می کند. پس خواهیم داشت :

$$63 * 0.15 \text{ sec} = 9.45 \text{ sec}$$