

whenever the ^{aggregate} avg. arrival rate exceeds the avg service rate, congestion occurs.

If packets may be dropped and delay will go up!

Earlier in Semester we saw that the Max # of bits that can be inserted into a pipe can't exceed the $BW \times Delay$ Product ^{if} RTT .

TCP Congestion Control

Tahoe

Reno

Tahoe Implementation

Slow Start
(ss)

Congestion
Avoidance

Congestion
Control.

SS

Sender Set his window

$$\omega_c = 1 \text{ MSS}$$

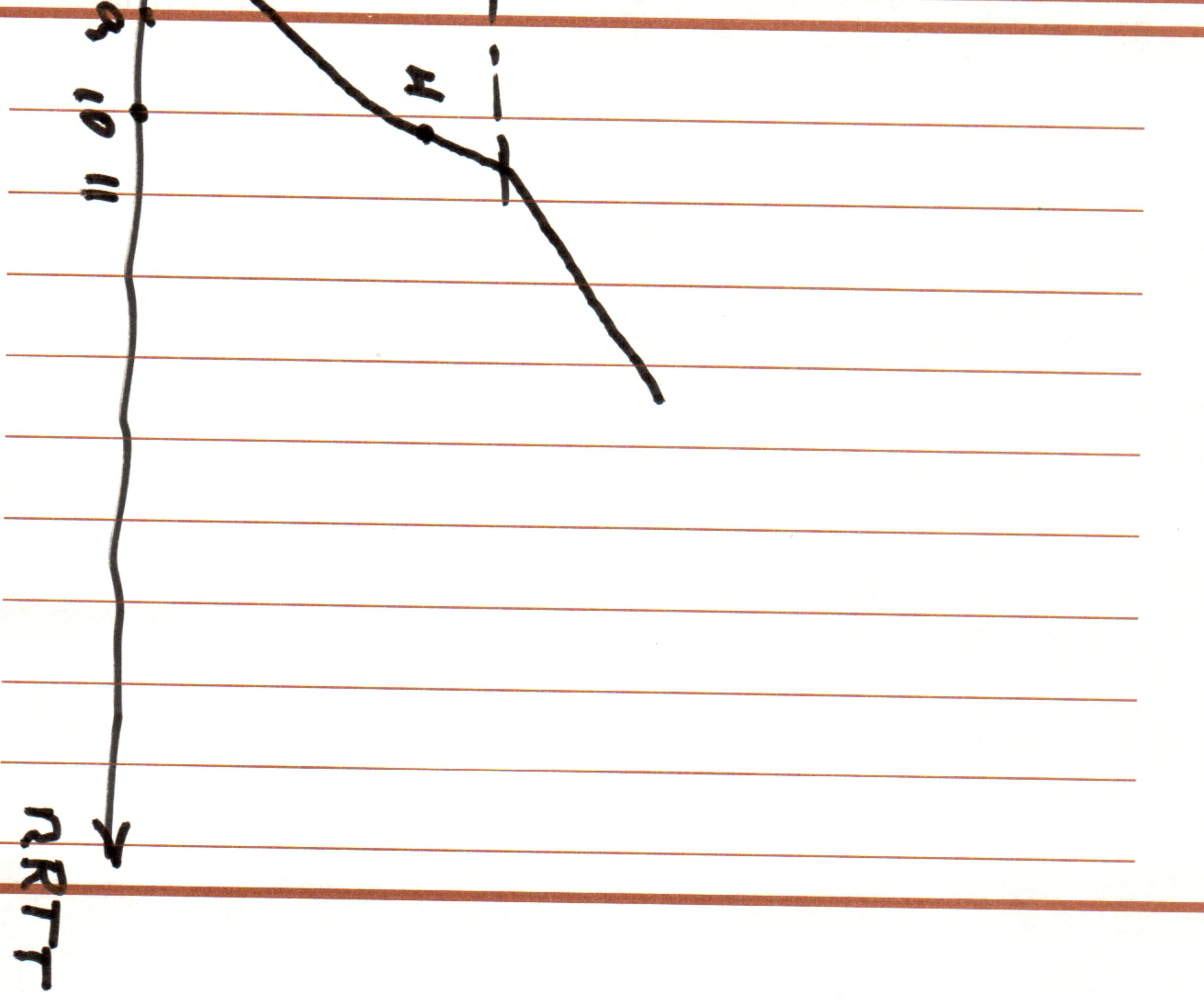
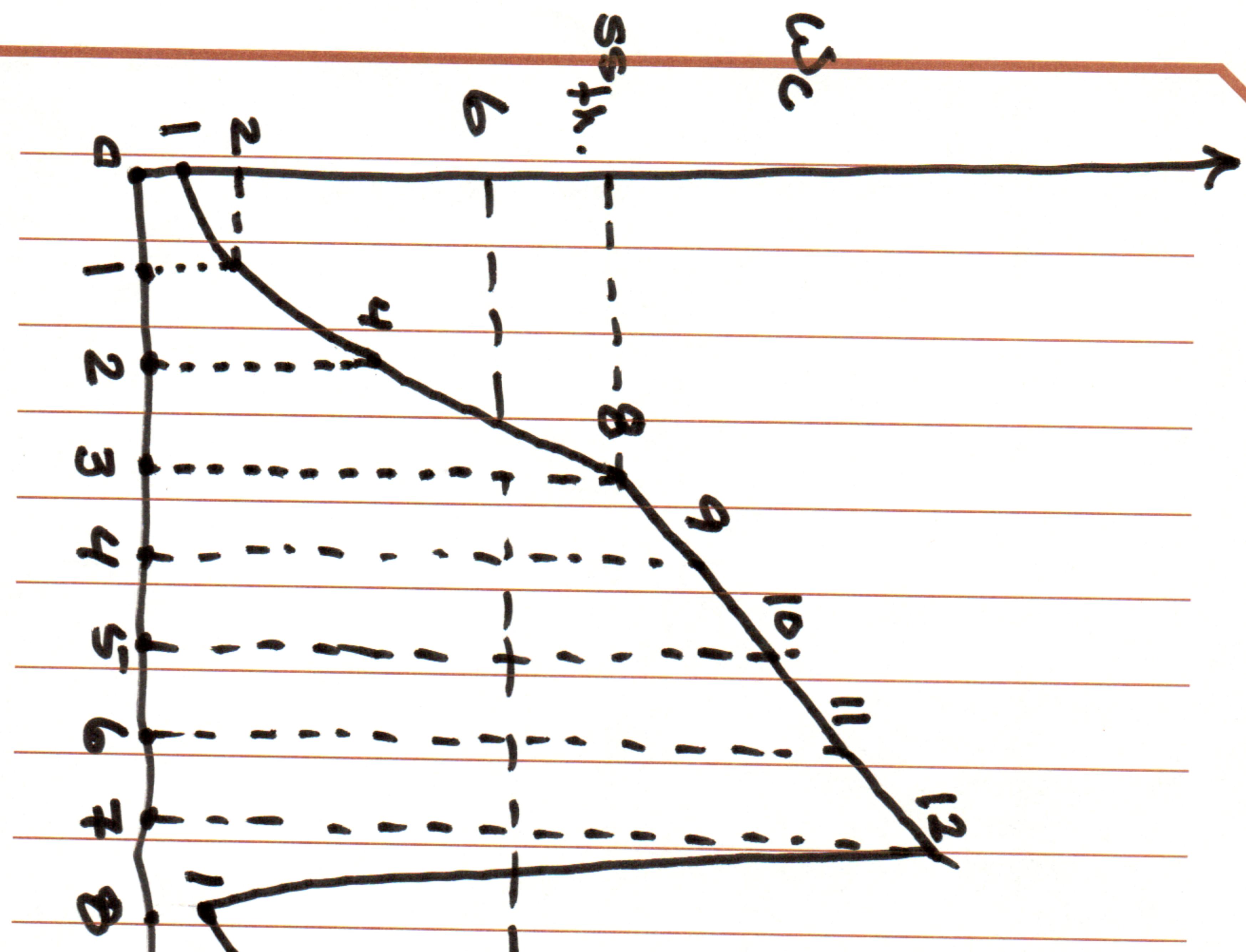
maximum
segment size
(Bytes)

If he receives

ACK (before Time-out expired)
he increment his window to
according to

$$\omega_c \leftarrow \omega_c + 1$$

↑ for every
req.
acted.



What is the Throughput at end of 1 round?

$$= \frac{1 \text{ MSS}}{1 \text{ RTT}} = 1 \frac{\text{Byte}}{\text{RTT}}$$

What is the total throughput at end of 2nd RTT?

$$= \frac{3 \text{ MSS}}{2 \text{ RTT}} = 1.5$$

" " " " at
end of 3rd RTT

$$= \frac{7 \text{ MSS}}{3 \text{ RTT}}$$

" " " " 4th round

$$= \frac{15 \text{ MSS}}{4 \text{ RTT}}$$

The sender specifies an SS threshold

When we reached ss threshold

Sender exit SS-phase and enter

(C A)

Congestion Avoidance Phase

Back to our exp. Suppose

$$\frac{SS}{threshold} = 8 \text{ MSS}$$

he will increment his window according

$$w_c \leftarrow w_c + 1$$

"for every RTT.

or equivalently

$$w_c \leftarrow w_c + \frac{1}{w_c}$$

↑
for every

Throughput at
end of 5th round
Aug. Ackd.

$$= \frac{24 \text{ MSS}}{SRT}$$

Congestion Control Phase

Back to our eq:

at beginning of 8th RTT, the Sender sent 12 MSS, at end of RTT, NO ACK was received.

RTO expired!

Sender enters Congestion Control phase

$$w_c \leftarrow 1$$

and he retransmit the segment that was timed-out and ~~not~~ enters the SS-

^{SS}
phase again and set a new
SS threshold = $\frac{w_c + \text{window size}}{2}$ before
congestion occurred.

Back to our eq.

$$\text{New SS threshold} = \frac{12}{2} = 6$$

Reno implementation of TCPCC.

It is identical to Tahoe when a Time-out occurs.

However, it doesn't treat the 3 dup ACKs and the expiration of a time-out the same way as the

expiration of a time-out.

(It) introduces a new state Reno "called Fast Recovery state"

