

CSC358H5: Principles of Computer Networking — Winter 2025

Worksheet 11 Solution: TCP and NAT

Q0 Knowledge Check

0.a Which of the following statements are true? (Select all that apply)

- ☒ Middleboxes may break IP design principles but are still considered necessities.
- ☐ Middleboxes are considered to be link-layer devices.
- ☐ In practice, most TCP connections provide congestion control, but not flow control.
- ☒ Congestion control ensures a group of senders do not overwhelm the network.

0.b Which one of the following statements are true? (Select all that apply)

- ☐ Fast retransmission is not effective when we have a large congestion window.
- ☒ Fast retransmission is not effective when packet drops are bursty.
- ☐ Fast retransmission is the exponential growth of congestion window when a new TCP connection is established.
- ☐ Fast retransmission starts after retransmission time-out expires.

0.c A Network Address Translator (NAT) has a binding timer associated with each map entry. Why is this necessary?

Answer. Eventually, the NAT box needs to reclaim memory and port numbers to use for new data transfers. Also, a stale map entry could be viewed as a security risk, as an attacker could send packets to the hosts behind the NAT box using a source address and port number of a past transfer.

0.d Why has Network Address Translation been so widely deployed, despite the intent that it was meant to provide temporary relief from IP address space exhaustion until IPv6 could be designed and deployed?

Answer. NAT is plug-and-play, whereas IPv6 requires upgrades to the hosts and routers to implement the new network-layer protocol. Also, NAT offers additional advantages, such as hiding multiple machines behind a single IP address allocated by a service provider.

Q1 Consider the following customized version of TCP which

- implements AIMD
- does NOT implement slow start mechanism
- starts with congestion window of 1 MSS
- for simplicity, assume that the first data byte starts with sequence number 0
- for simplicity, assume that the receiver drops any out of order sequences (*i.e.*, if it receives a packet which is out of order, the receiver will drop that packet and won't store it in its receiving buffer).

Moreover, assume that the sender and receiver are connected to each other with a direct link with infinite capacity. As a result, the sender can detect a packet loss after one RTT.

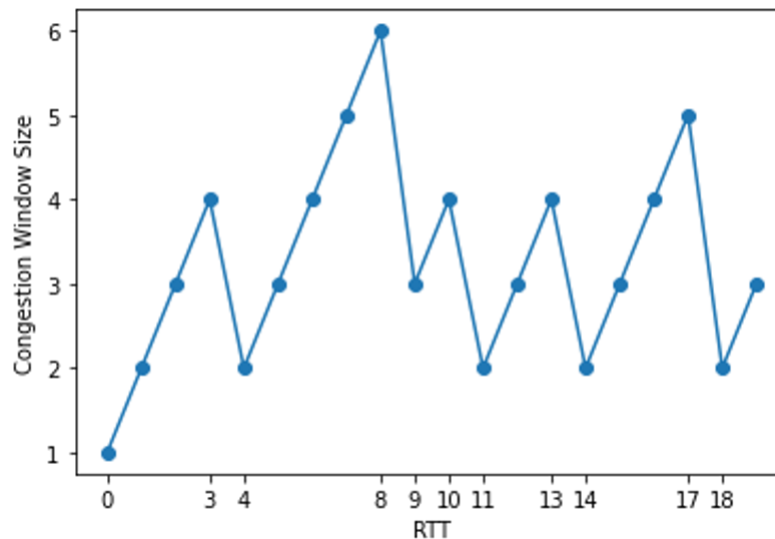
Assume that packets with sequence number 8 MSS, 24 MSS, 29 MSS, 37 MSS, and 49 MSS are dropped once.

1.a Draw the congestion window size during the first 12 RTT, *i.e.*, from $t = 0$ to $t = 12$ RTT.

Answer. This is how CWND changes in each RTT.

- At $t = 0$, $CWND = 1$ MSS and the sender sends packet with sequence number 1.
- At $t = 1 \times RTT$, the sender receives $ACK = 1$ MSS.
 - It changes congestion window size to $CWND = 2$ MSS.
 - It sends packets with sequence number 1 MSS and 2 MSS.
- At $t = 2 \times RTT$, the sender receives $ACK = 2$ MSS and $ACK = 3$ MSS.
 - It changes congestion window size to $CWND = 3$ MSS.

- It sends packets with sequence number 3 MSS and 4 MSS and 5 MSS.
- At $t = 3 \times \text{RTT}$, the sender receives $\text{ACK} = 4$ to 6 MSS.
 - It changes congestion window size to $\text{CWND} = 4$ MSS.
 - It sends packets with sequence number 6 to 9 MSS.
- At $t = 4 \times \text{RTT}$, the sender receives $\text{ACK} = 7$ MSS and $\text{ACK} = 8$ MSS and $\text{ACK} = 8$ MSS.
 - It detects that the packet with sequence number 8 MSS was dropped.
 - It halves its congestion window size to $\text{CWND} = 2$ MSS.
 - It sends packets with sequence number 8 MSS and 9 MSS.
- At $t = 5 \times \text{RTT}$, the sender receives $\text{ACK} = 9$ MSS and $\text{ACK} = 10$ MSS.
 - It increases its congestion window size to $\text{CWND} = 3$ MSS.
 - It sends packets with sequence number 10 to 12 MSS.
- At $t = 6 \times \text{RTT}$, the sender receives $\text{ACK} = 11$ to 13 MSS.
 - It increases its congestion window size to $\text{CWND} = 4$ MSS.
 - It sends packets with sequence number 13 to 16 MSS.
- At $t = 7 \times \text{RTT}$, the sender receives $\text{ACK} = 14$ to 17 MSS.
 - It increases its congestion window size to $\text{CWND} = 5$ MSS.
 - It sends packets with sequence number 17 to 21 MSS.
- At $t = 8 \times \text{RTT}$, the sender receives $\text{ACK} = 18$ to 22 MSS.
 - It increases its congestion window size to $\text{CWND} = 6$ MSS.
 - It sends packets with sequence number 22 to 27 MSS.
- At $t = 9 \times \text{RTT}$, the sender receives $\text{ACK} = 23$ and multiple $\text{ACK} = 24$ MSS.
 - It halves its congestion window size to $\text{CWND} = 3$ MSS.
 - It sends packets with sequence number 24 to 26 MSS.
- At $t = 10 \times \text{RTT}$, the sender receives $\text{ACK} = 25$ to 27 MSS.
 - It increases its congestion window size to $\text{CWND} = 4$ MSS.
 - It sends packets with sequence number 27 to 30 MSS.
- At $t = 11 \times \text{RTT}$, the sender receives $\text{ACK} = 28$ and multiple acknowledgments with ACK number 29 MSS.
 - It halves its congestion window size to $\text{CWND} = 2$ MSS.
 - It sends packets with sequence number 29 and 30 MSS.



1.b What is the throughput at the sender by the end of 12th RTT (*i.e.*, right before $t = 12$ RTT)?

Answer. The throughput is $T = \frac{(1+2+3+4+2+3+4+5+6+3+4+2)MSS}{12 \times RTT} = \frac{39 \times MSS}{12 \times RTT}$

1.c What is the goodput at the receiver by the end of 12th RTT (*i.e.*, right before $t = 12$ RTT)?

Answer. The goodput is $T^{\text{goodput}} = \frac{31 \times MSS}{12 \times RTT}$

Q2 Consider a TCP connection, which only implements AIMD and Fast Retransmission, *i.e.*, upon receiving a new ACK it increments CWND, and upon time-out or receiving triple duplicate ACKs it halves the CWND.

For simplicity, assume MSS is 1 byte, RTT is 1 second, and the transmission delay to send each segment is 0.1 seconds. Furthermore, assume that

- the last ACK sequence number that was received before $t = 1$ was 101.
- At $t = 0$, the CWND size was 10 MSS and the sender sent the packets with sequence numbers 101 to 110 at $t = 0, 0.1, \dots, 0.9$ second, respectively.
- the packet with sequence number #102 is lost only for its first transmission.
- the sender is can only transmit segment with size MSS, *i.e.*, it cannot send a segment with size less than 1 MSS.
- RTO is fixed and set as $3 \times RTT$.

2.a Fill in the tables below, until the sender transmits the packet with sequence number #116.

[NOTE: CWND will be incremented upon receiving an ACK with new acknowledgment number and won't be incremented if a duplicate ACK is received.]

[HINT: After receiving the first three duplicate ACKS with ACK number x , TCP Fast Retransmission mechanism will take action accordingly and any subsequent ACK with number x will be ignored.]

Time (sec)	Received ACK no.	CWND	Transmit Seq no.	Packets in flight
1.0	102 (101)	$10 + \frac{1}{10} = 10.1$	111	102 to 111
1.2	102 (103)	10.1	/	102 to 111
1.3	102 (104)	10.1	/	102 to 111
1.4	102 (105)	$\frac{10.1}{2} = 5.05$	102	102 to 111
1.5	102 (106)	5.05	/	102 to 111
1.6	102 (107)	5.05	/	102 to 111
1.7	102 (108)	5.05	/	102 to 111
1.8	102 (109)	5.05	/	102 to 111
1.9	102 (110)	5.05	/	102 to 111
2	102 (111)	5.05	/	102 to 111
2.4	112 (102)	$5.05 + \frac{1}{5.05} \approx 5.25$	112	112
2.5	/	5.25	113	112, 113
2.6	/	5.25	114	112-114
2.7	/	5.25	115	112-115
2.8	/	5.25	116	112-116

2.b Find the average throughput of this TCP during the time $t = 1.0$ to 3.0 second.

Answer. It is not difficult to see that the next transmission will occur at $t = 3.4$ seconds. Thus, the average throughput during the time $t = 1.0$ to 3.0 is $T = \frac{7}{3-1} = 3.5$ byte-per-second.